

Hydrodynamic Simulations in 3+1 General Relativity

Matthew D. Duez, Pedro Marronetti, and Stuart L. Shapiro*

Department of Physics, University of Illinois at Urbana-Champaign, Urbana, IL 61801

Thomas W. Baumgarte†

Department of Physics and Astronomy, Bowdoin College, Brunswick, ME 04011

We solve Einstein's field equations coupled to relativistic hydrodynamics in full 3+1 general relativity to evolve astrophysical systems characterized by strong gravitational fields. We model rotating, collapsing and binary stars by idealized polytropic equations of state, with neutron stars as the main application. Our scheme is based on the BSSN formulation of the field equations. We assume adiabatic flow, but allow for the formation of shocks. We determine the appearance of black holes by means of an apparent horizon finder. We introduce several new techniques for integrating the coupled Einstein-hydrodynamics system. For example, we choose our fluid variables so that they can be evolved without employing an artificial atmosphere. We also demonstrate the utility of working in a rotating coordinate system for some problems. We use rotating stars to experiment with several gauge choices for the lapse function and shift vector, and find some choices to be superior to others. We demonstrate the ability of our code to follow a rotating star that collapses from large radius to a black hole. Finally, we exploit rotating coordinates to evolve a corotating binary neutron star system in a quasi-equilibrium circular orbit for more than two orbital periods.

PACS numbers: 04.30.Db, 04.25.Dm, 97.80.Fk

I. INTRODUCTION

With the availability of unprecedented observational data, the physics of compact object is entering a particularly exciting phase. New instruments, including X-ray and γ -ray satellites and neutrino observatories, are detecting signals from highly relativistic events in regions of strong gravitational fields around neutron stars and black holes. A new generation of gravitational wave interferometers is promising to open a completely new window for the observation of compact objects. The ground-based gravity wave observatories LIGO and TAMA are already operational and are collecting data, GEO and VIRGO will be completed soon, and a space-based interferometer LISA is currently under design.

Given the small signal-to-noise ratio in these new gravitational wave detectors, theoretical models of likely sources are needed for the positive identification of the signal as well as for its physical interpretation [1]. One promising technique for the identification of signals in the noise output of the detector is matched filtering, which requires accurate theoretical gravitational wave templates [2]. The need for such templates has driven a surge of interest in developing reliable techniques capable of their construction.

Compact binaries, i.e. binaries consisting of either black holes or neutron stars, are among the most promising sources of gravitational radiation. Much progress has

been made in refining post-Newtonian point-mass approximations. These are suitable for large binary separations for which relativistic effects are sufficiently small and any internal structure can be neglected [3]. At small binary separations, the most promising technique for modeling the inspiral, coalescence and merger is numerical relativity.

Several other observed phenomena involving compact objects require numerical relativity for their modeling. One such example is Gamma Ray Bursts (GRBs). While it is not yet known what the origin of GRBs is, the central source is almost certainly a compact object [4]. Most scenarios involve a rotating black hole surrounded by a massive magnetized disk, formed by a supernova, or the coalescence of binary neutron stars [5]. To confirm or refute any GRB scenario requires numerical studies in full 3+1 relativistic magnetohydrodynamics.

Another astrophysical scenario requiring numerical treatment is the formation of supermassive black holes (SMBHs). Among the scenarios proposed to explain SMBH formation are the collapse of a relativistic cluster of collisionless matter, like a relativistic star cluster [6] or self-interacting dark matter halo [7], or the collapse of a supermassive star [8]. Depending on the details of the collapse, SMBH formation may generate a strong gravitational wave signal in the frequency band of the proposed space-based laser interferometer LISA. Understanding the SMBH formation route may shed key insight into structure and galaxy formation in the early universe.

Solving the coupled Einstein field and hydrodynamics equations is a challenging computational task, requiring the simultaneous solution of a large number of coupled nonlinear partial differential equations. In addition to all of the usual problems of numerical hydrodynamics –

*Department of Astronomy & NCSA, University of Illinois at Urbana-Champaign, Urbana, IL 61801

†Department of Physics, University of Illinois at Urbana-Champaign, Urbana, IL 61801

handling advection, shock discontinuities, etc – one encounters the problems inherent to numerical relativity. The latter include identifying a suitable formulation of Einstein’s field equations, enforcing a well-behaved coordinate system, and, if black holes are formed, dealing with spacetime singularities.

The construction of self-consistent numerical solutions to the coupled equations of relativistic hydrodynamics and gravitation dates back to the pioneering work of May and White in spherical symmetry [9] (see also [10] for a review). In one of the first attempts to perform numerical integrations in three spatial dimensions, Wilson, Mathews, and Marronetti [11, 12, 13] (see [14, 15] for later corrections) tackled the binary neutron star problem. They simplified Einstein’s field equations by assuming that the spatial metric remains conformally flat at all times. Their implementation of relativistic hydrodynamics was based on earlier work by Wilson [16] and used upwind differencing to handle advection and artificial viscosity to capture shocks. The first fully self-consistent relativistic hydrodynamics code, which treats the gravitational fields without approximation, was developed by Shibata [17]. This code, based on earlier work by Shibata and Nakamura [18], adopts a Van Leer hydrodynamics scheme [19, 20] and also employs artificial viscosity for shocks. This code has been used in various astrophysical applications, including the coalescence and merger of binary neutron stars [21, 22] and the stability of single, rotating neutron stars [23, 24, 25]. In an alternative approach, Font *et al* [26] implemented a more accurate high-resolution shock-capturing technique to solve the equations of relativistic hydrodynamics. This code has been used to study pulsations of relativistic stars [27].

In this paper, we report on the status and some astrophysical applications of our new 3+1 general relativistic hydrodynamics code. Our code, based on the so-called Baumgarte-Shapiro-Shibata-Nakamura (BSSN) formulation of Einstein’s equations [18, 28], has several novel features, including an algorithm that does not require the addition of a tenuous, pervasive atmosphere that is commonly used in Eulerian hydrodynamical codes, both Newtonian and relativistic. This “no atmosphere” algorithm proves to be very robust and eliminates many problems associated with the traditional atmospheric approach [29].

We treat 1D shocks, spherical dust collapse to black holes, and relativistic spherical equilibrium stars to demonstrate the ability of our code to accurately evolve the coupled field and hydrodynamic equations in relativistic scenarios. We then use the evolution of stable and unstable uniformly rotating polytropes as a testbed to determine which gauge conditions are best-behaved in the presence of strong-field matter sources with significant angular momentum. We introduce rotating coordinate systems and show that these can yield more accurate simulations of rotating objects than inertial frames. We demonstrate the ability of our code to hold accu-

rately stable differentially rotating stars in equilibrium. We also show that our code can follow the collapse of rapidly differentially rotating stars reliably until an apparent horizon appears, by which time the equatorial radius has decreased from its initial value by more than a factor of ten.

We then turn to simulations of binary neutron stars. We adopt initial data describing corotating $n = 1$ polytropes in quasi-equilibrium circular orbit, and evolve these data for over two orbital periods. In this paper we present results for one particular binary and discuss the effect of corotating frames as well as the outer boundaries. An extended study, including binary sequences up to the dynamically identified innermost stable circular orbit (ISCO), will be presented in a forthcoming paper [30].

This paper is organized as follows. Secs. II and III describe our method of evolving the field and hydrodynamic equations, respectively. Sec. IV summarizes the various gauge choices with which we experiment. Sec. V lists the diagnostics used to gauge the reliability of our simulations. Sec. VII describes several tests of our algorithm. Sec. VIII applies our formalism to evolve non-rotating, uniformly rotating, and differentially rotating polytropes. In Sec. IX sketches our binary neutron star calculations. Our results are summarized in Sec. X. Some details of our hydrodynamic scheme and the rotating frame formalism are presented in the appendices.

II. GRAVITATIONAL FIELD EVOLUTION

A. Basic Equations

We write the metric in the form

$$ds^2 = -\alpha^2 dt^2 + \gamma_{ij}(dx^i + \beta^i dt)(dx^j + \beta^j dt), \quad (1)$$

where α , β^i , and γ_{ij} are the lapse, shift, and spatial metric, respectively. The extrinsic curvature K_{ij} is defined by

$$(\partial_t - \mathcal{L}_\beta)\gamma_{ij} = -2\alpha K_{ij}, \quad (2)$$

where \mathcal{L}_β is the Lie derivative with respect to β^i . We choose geometrized units with $G = c = 1$ throughout, so Einstein’s field equations are

$$G_{\mu\nu} = 8\pi T_{\mu\nu}. \quad (3)$$

We use greek letters to denote spacetime indices, and latin letters for spatial indices. Using the above variables, the field equations (3) split into the usual 3+1 ADM equations [31]. These consist of the Hamiltonian constraint

$$R - K_{ij}K^{ij} + K^2 = 16\pi\rho, \quad (4)$$

the momentum constraint

$$D_j K_i^j - D_i K = 8\pi S_i, \quad (5)$$

and the evolution equation for K_{ij}

$$\begin{aligned} (\partial_t K_{ij} - \mathcal{L}_\beta K_{ij}) &= -D_i D_j \alpha \\ &\quad + \alpha(R_{ij} - 2K_{il}K^l_j + K K_{ij}) \\ &\quad - 8\pi(S_{ij} + \frac{1}{2}\gamma_{ij}(\rho - S^i_i)) \end{aligned} \quad (6)$$

in addition to (2). Here D , R_{ij} and R are the covariant derivative operator, the three-dimensional Ricci tensor and the scalar curvature associated with γ_{ij} . The matter source terms ρ , S_i , and S_{ij} are projections of the stress-energy tensor with respect to the unit normal n^α on the time slice

$$\begin{aligned} \rho &= n_\alpha n_\beta T^{\alpha\beta} \\ S_i &= -\gamma_{i\alpha} n_\beta T^{\alpha\beta} \\ S_{ij} &= \gamma_{i\alpha} \gamma_{j\beta} T^{\alpha\beta}. \end{aligned} \quad (7)$$

Since numerical implementations of the ADM equations typically develop instabilities after very short times, we use a reformulation of these equations that is now often referred to as the BSSN formulation [18, 28]. This reformulation consists of evolving the conformally related metric $\tilde{\gamma}_{ij}$, the conformal exponent ϕ , the trace of the extrinsic curvature K , the conformal traceless extrinsic curvature \tilde{A}_{ij} , and the conformal connection functions $\tilde{\Gamma}^i$ defined by

$$\gamma_{ij} = e^{4\phi} \tilde{\gamma}_{ij} \quad (8)$$

$$K_{ij} = e^{4\phi} (\tilde{A}_{ij} + \frac{1}{3} \tilde{\gamma}_{ij} K) \quad (9)$$

$$\tilde{\Gamma}^i = -\tilde{\gamma}^{ij}{}_{,j}, \quad (10)$$

where $\det(\tilde{\gamma}_{ij}) = 1$ and $\text{tr}(\tilde{A}_{ij}) = 0$. In terms of these variables, Eqs. (2) and (6) become

$$(\partial_t - \mathcal{L}_\beta) \tilde{\gamma}_{ij} = -2\alpha \tilde{A}_{ij} \quad (11)$$

$$(\partial_t - \mathcal{L}_\beta) \phi = -\frac{1}{6} \alpha K \quad (12)$$

$$\begin{aligned} (\partial_t - \mathcal{L}_\beta) K &= -\gamma^{ij} D_j D_i \alpha + \frac{1}{3} \alpha K^2 \\ &\quad + \alpha \tilde{A}_{ij} \tilde{A}^{ij} + 4\pi \alpha (\rho + S) \end{aligned} \quad (13)$$

$$\begin{aligned} (\partial_t - \mathcal{L}_\beta) \tilde{A}_{ij} &= e^{-4\phi} (-D_i D_j \alpha + \alpha(R_{ij} - 8\pi S_{ij}))^{TF} \\ &\quad + \alpha(K \tilde{A}_{ij} - 2\tilde{A}_{il} \tilde{A}^l_j) \end{aligned} \quad (14)$$

and

$$\begin{aligned} \partial_t \tilde{\Gamma}^i &= \partial_j (2\alpha \tilde{A}^{ij} + \mathcal{L}_\beta \tilde{\gamma}^{ij}) \\ &= \tilde{\gamma}^{jk} \beta^i{}_{,jk} + \frac{1}{3} \tilde{\gamma}^{ij} \beta^k{}_{,kj} - \tilde{\Gamma}^j \beta^i{}_{,j} \\ &\quad + \frac{2}{3} \tilde{\Gamma}^i \beta^j{}_{,j} + \beta^j \tilde{\Gamma}^i{}_{,j} - 2\tilde{A}^{ij} \partial_j \alpha \\ &\quad - 2\alpha \left(\frac{2}{3} \tilde{\gamma}^{ij} K_{,j} - 6\tilde{A}^{ij} \phi_{,j} - \tilde{\Gamma}^i{}_{jk} \tilde{A}^{jk} + 8\pi \tilde{\gamma}^{ij} S_j \right) \end{aligned} \quad (15)$$

(see [28] for the computation of the Lie derivatives.)

In terms of the BSSN variables, the constraint equations (4) and (5) become, respectively,

$$\begin{aligned} 0 = \mathcal{H} &= \tilde{\gamma}^{ij} \tilde{D}_i \tilde{D}_j e^\phi - \frac{e^\phi}{8} \tilde{R} \\ &\quad + \frac{e^{5\phi}}{8} \tilde{A}_{ij} \tilde{A}^{ij} - \frac{e^{5\phi}}{12} K^2 + 2\pi e^{5\phi} \rho, \end{aligned} \quad (16)$$

$$0 = \mathcal{M}^i = \tilde{D}_j (e^{6\phi} \tilde{A}^{ji}) - \frac{2}{3} e^{6\phi} \tilde{D}^i K - 8\pi e^{6\phi} S^i, \quad (17)$$

where $S^i = \tilde{\gamma}^{ij} S_j$. While the two constraints are identically zero for analytical solutions, they vanish only approximately in numerical calculations. Thus, the Hamiltonian and momentum constraint residuals \mathcal{H} and \mathcal{M} can be monitored as a code test during numerical evolution calculations. In the BSSN formulation, we also monitor the new constraint

$$0 = \mathcal{G}^i = \tilde{\Gamma}^i + \tilde{\gamma}^{ij}{}_{,j}. \quad (18)$$

B. Boundary Conditions

Like any other hyperbolic system, the Einstein field equations must be supplemented by initial conditions and boundary conditions to have a unique evolution. We adopt boundary conditions that follow from the assumption of asymptotic flatness, i.e. $g_{\alpha\beta} \rightarrow \eta_{\alpha\beta}$. In the asymptotic domain, monopole terms dominate in the longitudinal variables, so $\phi \propto r^{-1}$. The transverse fields will be dominated by outgoing gravitational waves, so $\tilde{\gamma}_{ij} - \eta_{ij} \propto f(t-r)r^{-1}$ and $\tilde{A}_{ij} \propto g(t-r)r^{-1}$, where f and g are unknown functions of retarded time. Note that r^{-1} is a special case of $f(t-r)r^{-1}$, so that ϕ , $\tilde{\gamma}_{ij}$, and \tilde{A}_{ij} all satisfy outgoing wave boundary conditions. The appropriate boundary conditions for K and $\tilde{\Gamma}^i$ depend on the gauge conditions used in the interior.

C. Numerical Implementation

We evolve Eqs. (11)-(15) using an iterative Crank-Nicholson scheme with one predictor step and two corrector steps [32]. In this algorithm a function f with time derivative \dot{f} is updated from its value f^n at timestep n to its value f^{n+1} at the next timestep $n+1$ a time ΔT later. In the explicit predictor step ${}^1f^{n+1} = f^n + \Delta T \dot{f}^n$, where \dot{f}^n is computed from quantities on timestep n , a “predicted” new value ${}^1f^{n+1}$ is found. In the following two corrector steps, ${}^2f^{n+1} = f^n + \Delta T(\dot{f}^n + {}^1\dot{f}^{n+1})/2$ and $f^{n+1} = {}^3f^{n+1} = f^n + \Delta T(\dot{f}^n + {}^2\dot{f}^{n+1})/2$, these predicted values are “corrected”. The final value f^{n+1} converges quadratically in ΔT . ΔT is set by the Courant factor: $C = \Delta T/\Delta x$, where Δx is the coordinate distance between adjacent gridpoints. We typically use $C = 0.5$. The code implementing this evolution scheme has been discussed elsewhere [28], so we will highlight here only the new features of our code.

We enforce the algebraic constraints $\det \tilde{\gamma}_{ij} = 1$ and $\text{tr}(\tilde{A}_{ij}) = 0$ as described in [33]. Also following [33], we replace the term $\frac{2}{3}\tilde{\Gamma}^i\beta^j_{,j}$ in Eq. (15) with the analytically equivalent $-(\tilde{\gamma}^{ij}_{,j} + \frac{1}{3}\tilde{\Gamma}^i)\beta^j_{,j}$. These changes have little effect on the evolutions described in this paper, but lead to significant improvements when treating black holes by excision boundary conditions [33].

We use second order centered differencing for all spatial derivatives in the field equations. We have not found it necessary to use upwind differencing for any derivatives. We did find, however, that the addition of some dissipation in the evolution equation for ϕ increases the stability of the code. This can be supplied by upwind differencing of the term which advects ϕ along the shift, and we have confirmed that this will indeed improve the stability. However, we have chosen instead to add the Hamiltonian constraint to the evolution equation for ϕ , as follows

$$(\partial_t - \mathcal{L}_\beta)\phi = -\frac{1}{6}\alpha K + c_H \mathcal{H}. \quad (19)$$

Here the parameter c_H is set between $0.02\Delta T$ and $0.06\Delta T$. $c_H \mathcal{H}$ is a diffusive term, with Courant condition given [34] by $2c_H \Delta T / (\Delta x)^2 \leq 1$, so making c_H proportional to ΔT is necessary in order to avoid an instability at high resolutions. It also provides dimensional consistency in (19). Using Eq. (19) offers the advantage of significantly decreasing the growth in the error of the Hamiltonian constraint (see Sec. IX for an example of this.) We note that the above is similar to one of the modifications of BSSN suggested in [35].

D. Implementation of Boundary Conditions

As discussed in Section (IIB), we use Sommerfeld boundary conditions for most of the field variables. That is, the value of a quantity f on the boundary at time t and distance r from the origin is

$$f(r, t) = \frac{r - \Delta r}{r} f(r - \Delta r, t - \Delta T), \quad (20)$$

where ΔT is the timestep and $\Delta r = \alpha e^{-2\phi} \Delta T$.

For the functions $\tilde{\Gamma}^i$ we have experimented with several boundary conditions. We find little sensitivity to the condition used; the best choice seems to be fixing $\tilde{\Gamma}^i$ at their initial values (zero, for most of the applications here).

III. RELATIVISTIC HYDRODYNAMICS

A. Basic Equations

We describe the matter source of the Einstein equations as a perfect fluid so that the stress-energy tensor

can be written

$$T_{\mu\nu} = (\rho_0 + \rho_0\epsilon + P)u_\mu u_\nu + P g_{\mu\nu}. \quad (21)$$

Here ρ_0 , ϵ , P , and u_μ are the rest-mass density, specific internal energy, pressure, and fluid four-velocity, respectively. We adopt a Γ -law equation of state

$$P = (\Gamma - 1)\rho_0\epsilon, \quad (22)$$

where Γ is a constant. For isentropic flow, this is equivalent to the polytropic relation

$$P = \kappa \rho_0^\Gamma, \quad (23)$$

where κ is a constant. In our simulations we encounter non-isentropic flow (due to shocks), and hence use equation (22).

The equations of motion follow from the continuity equation

$$\nabla_\mu(\rho_0 u^\mu) = 0 \quad (24)$$

and the conservation of stress-energy

$$T^{\mu\nu}_{;\nu} = 0. \quad (25)$$

Following [17], these equations can be brought into the form

$$\partial_t \rho_\star + \partial_i(\rho_\star v^i) = 0 \quad (26)$$

$$\partial_t e_\star + \partial_i(e_\star v^i) = 0 \quad (27)$$

$$\partial_t \tilde{S}_k + \partial_i(\tilde{S}_k v^i) = -\alpha e^{6\phi} P_{,k} - w h \alpha_{,k} \quad (28)$$

$$\begin{aligned} & -\tilde{S}_j \beta^j_{,k} + \frac{\alpha e^{-4\phi} \tilde{S}_i \tilde{S}_j}{2wh} \tilde{\gamma}^{ij}_{,k} \\ & - \frac{2\alpha h(w^2 - \rho_\star^2)}{w} \phi_{,k}, \end{aligned}$$

where $h = 1 + \epsilon + P/\rho_0$, $\rho_\star = \rho_0 \alpha u^0 e^{6\phi}$, $w = \rho_\star \alpha u^0$, $e_\star = (\rho_0 \epsilon)^{1/\Gamma} \alpha u^0 e^{6\phi}$, $\tilde{S}_k = \rho_\star h u_k$, and $v^i = u^i/u^0$ is the 3-velocity. The quantity w is determined by the normalization condition $u^\nu u_\nu = -1$, which can be written

$$w^2 = \rho_\star^2 + e^{-4\phi} \tilde{\gamma}^{ij} \tilde{S}_i \tilde{S}_j \left[1 + \frac{\Gamma e_\star^\Gamma}{\rho_\star (w e^{6\phi} / \rho_\star)^{\Gamma-1}} \right]^{-2}. \quad (29)$$

The perfect fluid given by Eq. (21) generates the following source terms for the ADM equations

$$\rho = h w e^{-6\phi} - P \quad (30)$$

$$S_i = e^{-6\phi} \tilde{S}_i \quad (31)$$

$$S_{ij} = \frac{e^{-6\phi}}{wh} \tilde{S}_i \tilde{S}_j + P \gamma_{ij} \quad (32)$$

We will only be considering systems where there is vacuum everywhere outside the star or stars. Therefore, the appropriate boundary condition on the matter flow is that no material should be flowing into the grid through the outer boundaries.

B. Numerical Implementation

We evolve the hydrodynamic variables using an iterative Crank-Nicholson scheme. This scheme is slightly different from the one used to update the field variables. In the corrector steps, instead of weighting f^n and f^{n+1} equally (i.e. $f^{n+1} = f^n + \Delta T(0.5f^n + 0.5f^{n+1})$), we make the evolution more implicit by setting $f^{n+1} = f^n + \Delta T(0.4f^n + 0.6f^{n+1})$. This makes the code slightly more stable.

As is often done in hydrodynamics codes [36], the updating of the fluid variables onto a new timestep is divided into two steps (“operator splitting”): the advection step (accounting for the advective terms on the left-hand sides of Eqs. (26)-(28)), and the source step (accounting for the right-hand sides of Eqs. (26)-(28)). Each step of a Crank-Nicholson update consists of applying first an advection substep and then a source substep. Our scheme for carrying out the advection substep is similar to the Van Leer scheme, and is discussed in detail in Appendix A. Since Eq. (26) has no sources, ρ_* is completely updated after it is advected. Following [29], we then use the updated ρ_* to complete the updating of e_* and \tilde{S}_k . It is shown in [29] that this gives improved behavior in Newtonian simulations of binary polytropes.

C. Artificial Viscosity

We handle shocks by adding quadratic artificial viscosity. This consists of adding a viscous pressure [17]

$$P_{Qvis} = \begin{cases} C_{Qvis} A (\delta v)^2 & \text{for } \delta v < 0, \\ 0 & \text{otherwise,} \end{cases} \quad (33)$$

where A is defined as $\frac{e_*^\Gamma}{(w e^{6\phi}/\rho_*)^{\Gamma-1}}$ and $\delta v = 2\partial_k v^k \Delta x$. Shock heating causes an increase in the local internal energy. Following [17], we change equation (27) to

$$\partial_t e_* + \partial_i (e_* v^i) = -(\rho_0 \epsilon)^{-1+1/\Gamma} \frac{P_{Qvis}}{\Gamma} \partial_k \left(\frac{w e^{6\phi} v^k}{\rho_*} \right) \quad (34)$$

We have also implemented linear artificial viscosity terms [37] that can be used to dissipate radial oscillations triggered in stars by the truncation error associated with finite differencing. The corresponding addition to the pressure is

$$P_{Lvis} = \begin{cases} -C_{Lvis} \sqrt{(\Gamma/n)\rho_* A} \delta v & \text{for } \delta v < 0, \\ 0 & \text{otherwise.} \end{cases} \quad (35)$$

Linear viscosity can be used at the beginning of a run to drive the initial data to dynamical equilibrium and later switched off. Figure 1 shows an example of how the radial oscillations can be quenched by linear viscosity. For this particular example, the P_{Lvis} was active only where the rest mass density exceeded a particular threshold value, to force this dissipative effect only

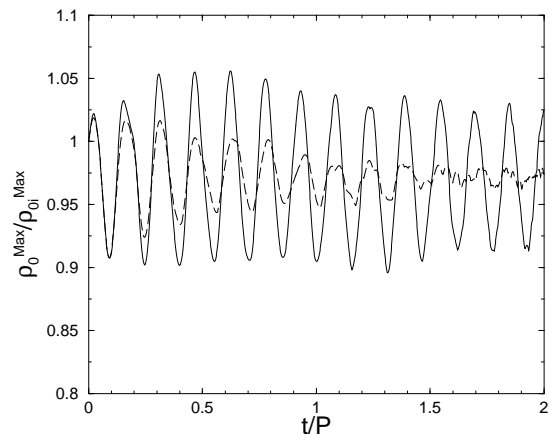


FIG. 1: Maximum rest mass density ρ_0 as a fraction of its initial value ρ_{0i} for the binary system shown in figure 19. Stellar radial oscillations can be efficiently quenched by the proper use of linear viscosity, as shown here. The solid line shows the evolution without linear artificial viscosity, while the dashed line shows the effect of this dissipative term.

deep inside the neutrons stars. The (small) dissipated kinetic energy goes into thermal energy. We typically use $0.1 \leq C_{Qvis} \leq 1.0$. Linear artificial viscosity is not used in the runs described below.

D. Non-Atmospheric Hydrodynamics

Numerical work in Eulerian hydrodynamics, both Newtonian and relativistic, has typically required the presence of a pervasive tenuous “atmosphere” that covers the computational grid outside the stars. To our knowledge, most published codes to date need to keep a minimum nonzero density that is usually set to be several orders of magnitude smaller than the maximum stellar density. Such an atmosphere has been necessary to prevent overflows arising from dividing by density in cells devoid of matter. This artificial atmosphere has to be small enough not to affect the true dynamical behavior of the system. However, very small values will propagate the round-off numerical error very quickly every time a division by the density is performed. A problem with the presence of this atmosphere is that as soon as the time evolution starts the material begins to fall onto the star, creating accretion shocks. Swesty *et al.* [29] solved this problem by adding a non-zero temperature to the atmosphere to restore some sort of equilibrium that would counterbalance the infall. Also, in order to avoid the bow shocks generated in the atmosphere by two stars in circular orbital motion, these authors provide the atmosphere with initial angular velocity. These are some of the typical problems present in the traditional artificial atmosphere approach found in many Eulerian hydrodynamics schemes.

In this paper we present a very simple algorithm that does not require the presence of atmospheric material. It consists of two ingredients. The first is the use of the spatial components of the linear momentum variable S_k as our hydrodynamical variable [38] instead of the traditional fluid four-velocity spatial components u^i used in most hydrodynamical codes (see for instance [17]). In the latter case, the Euler equation is used to update the flux $(\rho_\star u^i)$. Once this update is completed, the dynamical field u^i is recovered by dividing by the density ρ_\star . Using S_k as a variable, we avoid these divisions. The only time when the variable u^i needs to be calculated explicitly is when we need the three-velocity v^i that appears in every advection term on the left hand side of Eqs. (26-28). To avoid doing this calculation for very low values ρ_\star , we add the second ingredient: the introduction of a threshold value $\rho_\star \min$ below which all the hydrodynamical fields are set to vacuum values (i. e. $\rho_\star = u^i = v^i = 0$). A typical value for $\rho_\star \min$ is 10^{-7} times the maximum initial value of ρ_\star .

However, as the time evolution progresses, a tenuous shell of material typically drifts away from the stars and creates regions of very low density outside our stars. If nothing special is done about them, small shocks will heat this low-density region to very high temperatures and will generate large velocities. Although this low-density region has a negligible effect on our stars and spacetimes, it can cause the code to crash. Therefore, we impose a heating limit outside the star

$$e_\star = \min(e_\star, 10\rho_\star) \quad \text{if} \quad \rho_\star < e_{\text{factor}} \times \rho_{\star\text{max}}, \quad (36)$$

where e_{factor} is a constant that is determined empirically for a given physical scenario. We generally choose values between 10^{-3} and 10^{-6} , where the larger values of e_{factor} were only needed in simulations of collapsing stars with a strong bounce. We note that this is similar to the technique used in [26], in which the polytropic equation of state (23) is applied in the low-density region outside the star or stars.

E. Boundary Conditions

Since matter often diffuses outward, albeit in minute quantities, from the surface of the star(s) to the boundaries, we need to impose boundary conditions on the matter at the outer grid points. In algorithms where an artificial atmosphere is present, it is crucial to choose boundary conditions which do not lead to a continuous inflow from the boundary, or to bad behavior in the atmosphere. By eliminating such an atmosphere, however, all reasonable boundary conditions yield the same behavior so long as the boundaries are placed far enough from the star(s) that little matter ever reaches them.

We usually use an outflow boundary condition. For example, if the x -coordinate of gridpoints is indexed by an integer i with $i_{\min} \leq i \leq i_{\max}$, this boundary condition

at the outer- x boundary $i = i_{\max}$ is implemented as

$$\rho_{\star\text{imax}}^{n+1} = \rho_{\star\text{imax}-1}^{n+1} \quad (37)$$

$$e_{\star\text{imax}}^{n+1} = e_{\star\text{imax}-1}^{n+1} \quad (38)$$

$$\tilde{S}_{\text{imax}}^{n+1} = \begin{cases} \tilde{S}_{\text{imax}-1}^{n+1} & \text{if } \tilde{S}_{\text{imax}-1}^{n+1} > 0 \\ 0 & \text{otherwise} \end{cases} \quad (39)$$

We have experimented with other boundary conditions as well. We have tried fixing ρ_\star , e_\star , and \tilde{S}_k at their initial values. We have also tried simply copying the adjacent gridpoint onto the boundary with no outflow restrictions (*Copy*). These conditions produce similar results to those of the outflow condition for all applications, while being somewhat less computationally expensive.

IV. GAUGE CHOICES

A. Lapse

We experiment with several time slicing conditions. First, we try maximal slicing, which enforces $K = \partial_t K = 0$

$$\text{mx:} \quad 0 = -\gamma^{ij} D_j D_i \alpha + \alpha \tilde{A}_{ij} \tilde{A}^{ij} + 4\pi\alpha(\rho + S) . \quad (40)$$

This slicing condition has the advantages of controlling K and avoiding singularities. Unfortunately, it is a computationally expensive gauge choice, since it involves solving an elliptic PDE every timestep. Therefore, we also try a slicing condition which approximates maximal slicing, the so-called “K-driver” proposed by Balakrishna *et al* [39]. The idea is to convert the elliptic equation (maximal slicing) into a parabolic evolution equation

$$\text{Kdr:} \quad \partial_t \alpha = -\epsilon(\partial_t K + cK) , \quad (41)$$

where ϵ and c are positive constants. The equation $\partial_t K = -cK$, corresponding to exponential decay in K , is the solution of equation (41) as $\epsilon \rightarrow \infty$. However, setting ϵ at too large a value in our code will produce a numerical instability. (See the discussion of c_H in Sec. II C.) Fortunately, this limitation can be overcome. We are able to effectively evolve with larger ϵ by breaking up each timestep into several substeps and evolve Eq.(41) using a smaller ΔT than that used by the other variables. On each substep, we use the values of the metric on the destination time level, so the process is equivalent to solving the elliptic equation $\partial_t K + cK = 0$ by relaxation, except that we do not carry the process to convergence. Instead, we typically use 5 substeps per step, with $\epsilon = 0.625$ and $c = 0.1$.

An even less computationally expensive lapse condition is harmonic slicing, which for vanishing shift reduces to

$$\text{hm:} \quad \partial_t(\alpha^{-1}\gamma^{-1/2}) = 0 . \quad (42)$$

We apply this condition unchanged for vanishing and non-zero shift, and find that it often gives behavior similar to that obtained by using the above two slicings.

B. Shift

We also experiment with different spatial gauge choices. The simplest admissible shift choice, which turns out to be surprisingly good for collapsing star applications, is to keep the shift “frozen” at its initial values

$$\text{fz: } \beta^i(t) = \beta^i(0) \quad (43)$$

at each grid point.

We also try the approximate minimal distortion (AMD) gauge introduced by Shibata [40]

$$\text{AMD: } \delta_{ij} \nabla^2 \beta^i + \frac{1}{3} \beta^k{}_{,kj} = J_j \quad (44)$$

where ∇^2 is the flat-space Laplacian and

$$J_i = 16\pi\alpha S_i + 2\tilde{A}_{ij}(\alpha^{,j} - 6\alpha\phi^{,j}) + \frac{4}{3}\alpha K_{,i}. \quad (45)$$

This gauge condition was designed to approximate the Smarr and York minimal distortion shift condition [41], which in turn was constructed to minimize gauge-related time variation in the spatial metric.

As Shibata points out [24], the AMD condition must be modified in the event of a collapse in order to prevent a “blowing out” of coordinates on the black hole throat, which manifests itself by a growth in the proper 3-volume element, i.e. by growth in ϕ . The blowing out can be controlled by preventing the radial component of the shift from becoming large and positive

$$\text{MAMD: } \beta^i = \begin{cases} \beta_{\text{AMD}}^i & \text{for } \phi_c < (4/3)\phi_{ci} \\ \beta_{\text{AMD}}^i - f\beta_{\text{AMD}}^r \frac{x^i}{r} & \text{otherwise} \end{cases}, \quad (46)$$

where β_{AMD}^i is the solution of equation (44), ϕ_c is the value of ϕ at the coordinate origin, $\phi_{ci} = \phi_c(t=0)$, and

$$f = \left(\frac{3\phi_c}{2\phi_{ci}} - 2 \right) \frac{1}{1 + (r/R)^4} \quad (47)$$

$$\beta_{\text{AMD}}^r = x^k \beta_{\text{AMD}}^k / r, \quad (48)$$

where R is a constant. This correction is only useful in configurations with near spherical symmetry, so that the collapse is nearly radial at the center. It is disabled for simulations of binary systems.

Finally, we try approximating the “Gamma-freezing” condition $\partial_t \tilde{\Gamma}^i = 0$ using a “Gamma-driver”, which enforces controls $\tilde{\Gamma}^i$ in the same way that the K-driver controls K

$$\text{Gdr: } \partial_t \beta^i = k(\partial_t \tilde{\Gamma}^i + \eta \tilde{\Gamma}^i). \quad (49)$$

Here k and η are positive constants, and, as with the K-driver, we can effectively make k larger than would otherwise be possible by breaking up each step into multiple substeps. This shift condition has been used successfully in black hole evolution calculations [42]. The

Gamma-freezing condition is closely related to minimal distortion (and hence approximate minimal distortion), and it is hence not surprising that the modification (46) must also be applied to the Gamma-driver shift. Typical values for the Gamma-driver’s parameters are $k = 0.01$ and $\eta = 0.2$, using 10 substeps per step.

We have also implemented of the full Gamma freezing condition ($\partial_t \tilde{\Gamma}^i = 0$). However, applying this condition requires solving three coupled elliptic equations each timestep (see Eq. (15)), and we have found Gamma freezing to be too computationally expensive to be worth solving exactly.

C. Rotating Frames

Rotating coordinate frames possess superior angular momentum conservation capability over inertial frames in many applications, such as the hydrodynamical evolution of binaries systems. In transforming from an inertial frame with coordinates $(\bar{t}, \bar{x}, \bar{y}, \bar{z})$ to a rotating frame with coordinates (t, x, y, z) and constant angular frequency $\vec{\Omega} = \Omega \vec{e}_z$, we apply the following relations

$$\begin{aligned} \bar{t} &= t \\ \bar{x} &= x \cos(\Omega t) - y \sin(\Omega t) \\ \bar{y} &= x \sin(\Omega t) + y \cos(\Omega t) \\ \bar{z} &= z, \end{aligned} \quad (50)$$

where the barred variables will represent quantities in the inertial frame in the remainder of this section. It is convenient to compare variables in the two frames at an instant $\bar{t} = t = 0$ at which the two frames are aligned. At this instant, the line element transforms from

$$d\bar{s}^2 = -(\bar{\alpha} - \bar{\beta}^i \bar{\beta}_i) d\bar{t}^2 + 2\bar{\beta}_i d\bar{x}^i d\bar{t} + \bar{\gamma}_{ij} d\bar{x}^i d\bar{x}^j$$

to

$$\begin{aligned} ds^2 = & - \left(\bar{\alpha} - \bar{\gamma}_{ij} (\bar{\beta}^i + (\vec{\Omega} \times \vec{r})^i) (\bar{\beta}^j + (\vec{\Omega} \times \vec{r})^j) \right) dt^2 \\ & + 2\bar{\gamma}_{ij} (\bar{\beta}^i + (\vec{\Omega} \times \vec{r})^i) dx^j dt + \bar{\gamma}_{ij} dx^i dx^j, \end{aligned}$$

where $\vec{r} \equiv (x, y, z)$. From this equation, we see that the following transformation rules apply at $\bar{t} = t = 0$:

$$\begin{aligned} \alpha &= \bar{\alpha} \\ \beta^i &= \bar{\beta}^i + (\vec{\Omega} \times \vec{r})^i \\ \gamma_{ij} &= \bar{\gamma}_{ij}, \end{aligned} \quad (51)$$

Eq. (51) provides the transformation rules for the initial metric data from the inertial frame (where it is usually derived) to the rotating frame. The only change is the addition of a new term in the shift. At later times, vectors and tensors in the two frames will also differ by a rotation. However, we note that at all times there will be some inertial frame, related to $(\bar{t}, \bar{x}, \bar{y}, \bar{z})$ by a rotation matrix, which has axes aligned with the rotating frame and whose

metric is related to that of the rotating frame by Eq. (51). Using the coordinate transformations (50) we can derive the relation between all the fields in both frames, for example

$$\begin{aligned} u^0 &= \bar{u}^0 \\ u^i &= \bar{u}^i - (\vec{\Omega} \times \vec{r})^i \bar{u}^0 \\ u_i &= \bar{u}_i \\ v^i &= \bar{v}^i - (\vec{\Omega} \times \vec{r})^i, \end{aligned} \quad (52)$$

where v^i is the fluid three-velocity $v^i \equiv u^i/u^0$. At $t = \bar{t} = 0$, the components of any spatial tensor are unchanged under this transformation, for example

$$\gamma_{ij} = \bar{\gamma}_{ij}, \quad (53)$$

and equivalently for T_{ij} and K_{ij} . The relation (53) implies $\gamma^{ij} = \bar{\gamma}^{ij}$, since the inverse of a tensor is unique, so that we also find $K^{ij} = \bar{K}^{ij}$. To complete our introduction of rotating frames in general relativity, we refer the reader to Appendix B, where we show that the Newtonian limit of the relativistic Euler equation with a shift vector of the form of equation (51) reduces to the familiar Newtonian form of the Euler equation in rotating frames. In Appendix C, we show that the integrands used to evaluate M , M_0 , and J in Eqs. (63), (64), and (65), respectively, remain unchanged when expressed in terms of rotating frame variables.

D. Boundary Conditions

We always choose initial data which satisfy maximal slicing (40) and gauge choices which approximately maintain this slicing. Far from the source, equation (40) becomes the Laplace equation, and its solution can be written as a sum of multipole moment fields. In the presence of matter, the source will always have a nonzero monopole moment, so the asymptotic form of the lapse is

$$\alpha - 1 \propto r^{-1}. \quad (54)$$

All of our spatial gauge choices resemble one another, so we will just derive the shift boundary condition for the AMD shift (44), which is the easiest. The three components of Eq. (44) can be decoupled by decomposing β^i as in [40]

$$\beta^i = \delta^{ji} \left[\frac{7}{8} P_i - \frac{1}{8} (\eta_{,i} + P_{k,i} x^k) \right], \quad (55)$$

where x^k are the Cartesian coordinates. Eq. (44) then becomes

$$\nabla^2 P_i = J_i \quad (56)$$

$$\nabla^2 \eta = -J_i x^i. \quad (57)$$

To lowest order, $J_i = \rho v^i$. We will be studying systems with azimuthal velocity fields, for which $v^z = 0$, and

hence $\eta = P_z = 0$. The lowest nonvanishing moment of J_i , from the monopole piece of ρ , is $l = 1$, $m = \pm 1$. We can solve the Laplace equation (outside the star) assuming asymptotic flatness to get the boundary conditions $\beta^x \propto yr^{-3}$, $\beta^y \propto xr^{-3}$, and, to this order, $\beta^z = 0$. A nonzero boundary condition for β^z must come from a higher-order term, but, since β^z will be very small, our simulations are insensitive to it. We use

$$\beta^x \propto yr^{-3}, \quad \beta^y \propto xr^{-3}, \quad \beta^z \propto xyzr^{-7}. \quad (58)$$

The β^z condition is obtained by ignoring η and solving (56) subject to the lowest-order moment of the $A_{ij}\alpha^{,j}$ term in J_i (see Eq. (45).)

Note that the coordinate-rotation component of a shift, $\vec{\Omega} \times \vec{r}$, is a homogeneous solution of equation (44). It was eliminated in (58) by the assumption of asymptotic flatness. When working in a rotating frame, the shift does not obey an asymptotic flatness condition. Indeed, one can think of a coordinate rotation as a boundary condition imposed on the shift. In such frames, the asymptotic form of the shift is $\vec{\Omega} \times \vec{r}$ *plus* a piece which behaves like (58), and the boundary conditions must be set accordingly.

V. DIAGNOSTICS

In order to gauge the accuracy of our simulations, we monitor the L2 norms of the violation in the constraint equations. These are the Hamiltonian constraint \mathcal{H} (16), the momentum constraint \mathcal{M}^i (17), and the Gamma constraint (18). We normalize the Hamiltonian and momentum constraint violation by their L2 norm by

$$\begin{aligned} N_{HC} = & \left\| \left((2\pi\psi^5\rho)^2 + (\tilde{D}^i\tilde{D}_i\psi)^2 + \left(\frac{\psi}{8}\tilde{R}\right)^2 \right. \right. \\ & \left. \left. + \left(\frac{\psi^5}{8}\tilde{A}_{ij}\tilde{A}^{ij}\right)^2 + \left(\frac{\psi^5}{12}K^2\right)^2 \right)^{1/2} \right\|_2 \end{aligned} \quad (59)$$

and

$$\begin{aligned} N_{MC} = & \left\| \left(\sum_{i=1}^3 \left[(8\pi S^i)^2 + \left(\frac{2}{3}\tilde{D}^i K\right)^2 \right. \right. \right. \\ & \left. \left. + \left(\psi^{-6}\tilde{D}_j(\psi^6\tilde{A}^{ij})\right)^2 \right] \right)^{1/2} \right\|_2. \end{aligned} \quad (60)$$

The two terms in the Gamma constraint (18) often vanish individually, so that a similar normalization is not meaningful for this constraint.

Related to the Hamiltonian and momentum constraints are mass and angular momentum conservation. In Cartesian coordinates, the ADM mass M and the angular momentum J^i are defined by the behavior of the metric on a closed surface at asymptotically flat spatial

infinity

$$M = \frac{1}{16\pi} \int_{r=\infty} \sqrt{\gamma} \gamma^{im} \gamma^{jn} (\gamma_{mn,j} - \gamma_{jn,m}) d^2 S_i \quad (61)$$

$$J_i = \frac{1}{8\pi} \varepsilon_{ij}{}^k \int_{r=\infty} x^j K_k^m d^2 S_m. \quad (62)$$

Since Eq. (62) is computed at spatial infinity, $\varepsilon_{ij}{}^k$ is the flat-space Levi-Civita tensor. Using Gauss' Law, we transform the surface integrals into volume integrals

$$M = \int_V \left(e^{5\phi} (\rho + \frac{1}{16\pi} \tilde{A}_{ij} \tilde{A}^{ij} - \frac{1}{24\pi} K^2) \right. \quad (63)$$

$$\left. - \frac{1}{16\pi} \tilde{\Gamma}^{ijk} \tilde{\Gamma}_{jik} + \frac{1 - e^\phi}{16\pi} \tilde{R} \right) d^3 x$$

$$J_i = \varepsilon_{ij}{}^k \int_V \left(\frac{1}{8\pi} \tilde{A}_k^j + x^j S_k \right. \quad (64)$$

$$\left. + \frac{1}{12\pi} x^j K_{,k} - \frac{1}{16\pi} x^j \tilde{\gamma}^{lm}{}_{,k} \tilde{A}_{lm} \right) e^{6\phi} d^3 x$$

(see, for example, Appendix A in [33] for a derivation). Note that since $\varepsilon_{ij}{}^k$ is outside the integral, it is still the flat-space Levi-Civita tensor. Baryon conservation $(\rho_0 u^\mu)_{;\mu} = 0$ implies that the rest mass

$$M_0 = \int \rho_\star d^3 x \quad (65)$$

is also conserved. Due to the finite differencing in our hydrodynamic scheme, M_0 is conserved identically except for matter flow off the computational grid. We therefore monitor M_0 only as a diagnostic of how much matter flows through the outer boundaries.

Eqs. (63) and (64) are only valid in asymptotically flat spatial hypersurfaces and thus are not suited for use in rotating reference frames. However, the problem can be sidestepped quite easily by calculating the mass and angular momentum in the inertial frame, as functions of the dynamical variables of the rotating frame. In Appendix C we show that the integrals for these conserved quantities are exactly the same when expressed in terms of the rotating frame fields.

Another useful quantity to monitor is the circulation. According to the Kelvin-Helmholtz theorem [43], the relativistic circulation

$$\mathcal{C}(c) = \oint_c h u_\mu \lambda^\mu d\sigma \quad (66)$$

is conserved in isentropic flow along an arbitrary closed curve c when evaluated on hypersurfaces of constant proper time. Here $h = 1 + \varepsilon + P/\rho$ is the specific enthalpy, σ is a parameter which labels points on c , and λ^μ is the tangent vector to the curve c . Since $\mathcal{C}(c)$ is only conserved for isentropic flow, checking conservation of circulation along a few curves will measure the importance of numerical and artificial viscosity on an evolution. We do not monitor circulation in this paper, although such a check has been implemented elsewhere [44].

Finally, we check for the existence of apparent horizons using the apparent horizon finder described in [45].

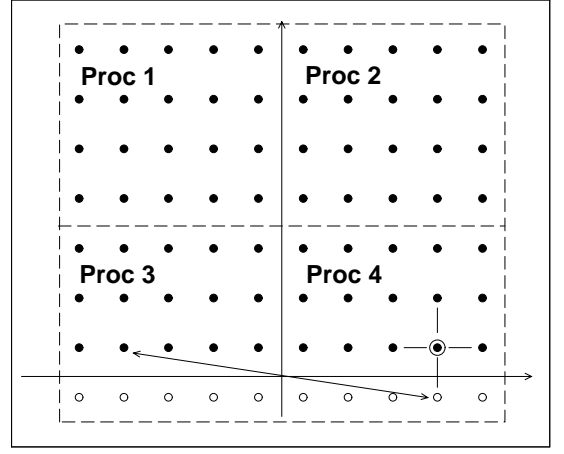


FIG. 2: This diagram shows how our code implements π -symmetry in distributed-memory computer clusters. The black circles correspond to grid points, and the bottom row corresponds to the boundary in the plane orthogonal to the rotation axis. The white circles represent the ghostzones needed by our second order finite difference stencil. The arrow connects two points that are related in the presence of π -symmetry.

VI. NUMERICAL CODE DESCRIPTION

All our algorithms have been implemented in a parallel, distributed-memory environment using DAGH software [46] developed as part of the Binary Black Hole Grand Challenge Alliance. When we need to solve elliptic equations (to construct initial data or to impose elliptic gauge conditions), we use the computational toolkit PETSc [47].

Due to our large number of variables, the memory needed by our code is considerable (for example, a run with 64^3 spatial zones may require up to 2 Gbytes of memory). Thus, it is crucial that we exploit any symmetries present in a given problem to minimize the number of gridpoints needed. We have implemented reflection symmetry across a coordinate plane (equatorial symmetry) and reflection symmetry across three coordinate planes (octant symmetry), which cut the size of our grids by factors of two and eight. Our code also allows us to enforce π -symmetry, which assumes symmetry under a rotation of π radians about a given axis. Unlike equatorial and octant symmetry, the implementation of π -symmetry is not trivial on distributed-memory parallel systems. This is because grid points needed to generate the proper boundary conditions at a given location of the outer grid boundary will usually be located in the memory of a different processor, as seen in the diagram of Fig. 2 where the value of the field at the white circle needed by point P in processor 4 must be provided by a black circle on processor number 3. We fix this problem by creating a two dimensional array for each field that stores the values of the field on all the grid points out-

side the boundary (white circles) needed to calculate the derivatives of the field at the grid points at the boundaries (first row of black points). Each processor is responsible for updating the array values corresponding to grid points within its domain by a π -radian rotation. Updated values are broadcast via MPI, and each processor has a copy of the complete two-dimensional array from which to draw the corresponding boundary values.

VII. TESTS

A. Vacuum Code Tests

The algorithm for evolving the field equations was first tested in the context of small amplitude gravitational waves [28]. With harmonic slicing, the system could be accurately evolved for over 100 light crossing times without any sign of instability. The results also showed second-order convergence to the analytic solution when resolution was increased. In a forthcoming paper [33], it will be demonstrated that this same code can stably evolve isolated black holes, with and without rotation, in Kerr-Schild coordinates.

B. Hydro-without-Hydro

Next, it was demonstrated that this field evolution scheme is stable when predetermined matter sources are present [48]. This was done by inserting the matter sources from known solutions of the Einstein equations and then evolving the gravitational field equations. Using this “hydro-without-hydro” approach, [48] evolved the Oppenheimer-Volkoff solution for static stars without encountering any instability, and the Oppenheimer-Snyder solution for collapse of homogeneous dust spheres well past horizon formation. The same hydro-without-hydro approach was later used to model the quasi-equilibrium inspiral of binary neutron star systems and calculate the complete late-inspiral gravitational wavetrain outside the ISCO [49, 50].

C. Shock Tube

Every hydrodynamic algorithm must demonstrate some ability to handle shocks. In Fig. 3, we compare the output of our code for a simple one-dimensional shock tube problem with the exact result, which is known analytically in special relativity [51]. In order to compare with this result, the metric functions are held at their Minkowski values throughout this test. At $t = 0$, we set $v \equiv v^x = 0$ everywhere. For $x < 0$ we set $\rho_0 = 15$, $P = 225$ initially, and for $x > 0$ we set $\rho_0 = 1$, $P = 1$. We output data at $t = 0.5$. In Figure 1, we use artificial viscosity parameters $C_{Qvis} = 1$, $C_{Lvis} = 0$ (see Sec. III C) and a grid with 400 points. The shock is resolved quite

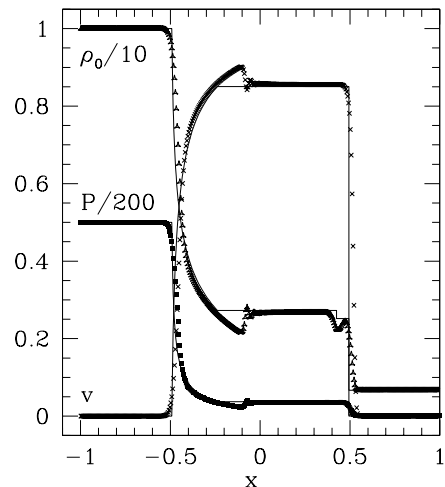


FIG. 3: The one-dimensional relativistic Riemann shock tube test. We plot the numerical rest density ρ_0 (triangles), pressure P (squares), and velocity v (crosses) at $t = 0.5$. Solid curves show the analytic values. This particular run used $C_{Qvis} = 1$.

well, and the only disturbing feature of our results is the “overshoot” in variables at the rarefaction wave. Norman and Winkler [52] have shown that these overshoots are present in the solution to the finite difference equations of artificial viscosity schemes even in the limit of the grid spacing going to zero. This problem therefore represents a fundamental limitation of artificial viscosity schemes, and points to the need for more sophisticated high-resolution-shock-capturing techniques when strong shocks are present (see, e.g., [26]). However, for many of our astrophysical applications (e.g. binary inspiral) we anticipate at most very weak shocks, so that the use of artificial viscosity schemes is adequate.

Our results are completely insensitive to C_{Qvis} when it is within the range $0 - 0.1$. We find the optimal behavior around $C_{Qvis} \approx 1$, at which point the effects of artificial viscosity are small but noticeable. For $C_{Qvis} \approx 5$ or greater, the viscosity is too large, and we are unable to evolve accurately.

Note that in the above example $\rho_\star > 10^{-2}\rho_{\star\max}$ everywhere, so e_\star -limiting (36) is never used. More extreme shocks can be created by increasing the density ratio $\rho_0(x > 0)/\rho_0(x < 0)$. We find that we can treat shocks reasonably accurately for ratios of up to about twenty.

D. Oppenheimer-Snyder dust collapse

As a second simulation which can be tested against exact results, we model Oppenheimer-Snyder (OS) collapse of a homogeneous dust sphere to a black hole [53]. The

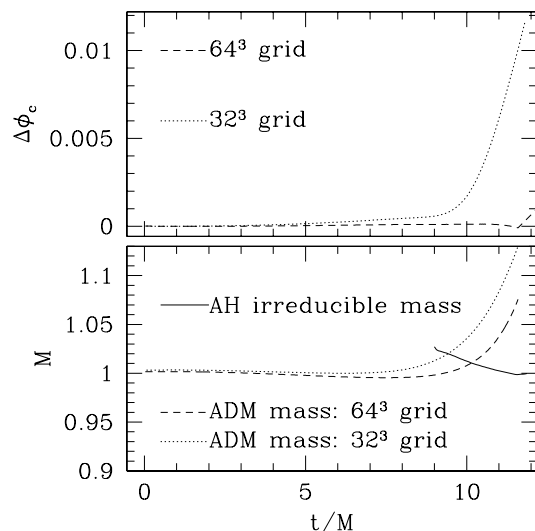


FIG. 4: The evolution of the conformal exponent at the origin ϕ_c and the ADM mass M during Oppenheimer-Snyder collapse. The deviation of ϕ_c from its analytic value ϕ_c^{anal} is measured by $\Delta\phi_c = (\phi_c - \phi_c^{\text{anal}})/(\phi_c + \phi_c^{\text{anal}})$. This is plotted on the top panel. On the bottom panel, we plot the ADM mass of the system and the irreducible mass of black hole, given by the area of the apparent horizon. We compare runs at two different resolutions.

analytic solution for OS collapse can be transformed into maximal slicing and isotropic coordinates following [54]. We use the analytic solution at $t = 0$, when the matter is at rest, as initial data for all variables. We then evolve the gravitational and hydrodynamic fields with our 3+1 code and compare the result with the exact solution. At each timestep, we determine the lapse by solving the maximal slicing condition from the fields on our 3D grid. For the shift we insert the analytic values corresponding to isotropic coordinates. We evolve on a 32^3 grid and a 64^3 grid, utilizing octant symmetry to treat only the upper octant. Our outer boundaries are placed at $4M$ in the isotropic coordinates of our grid. The initial Schwarzschild radius of the dust sphere is $3M$.

In Fig. 4, we show the convergence of the central conformal exponent ϕ_c to the exact value. In Fig. 5, we compare the density profiles at several times for the 64^3 grid to their analytical values. Throughout the evolution, we search for apparent horizons. At $t = 8.75M$, we locate an apparent horizon with irreducible mass $M_{\text{AH}}/M = 1.03$. (See Fig. 4.) This mass remains constant to within 3% until the end of the simulation ($\approx 3M$ later). As is known analytically, all of the mass falls inside the black hole.

This test is similar to the “hydro-without-hydro” Oppenheimer-Snyder test performed on our code in [48], except that here the matter fields and the lapse are determined numerically rather than set to their analytic values.

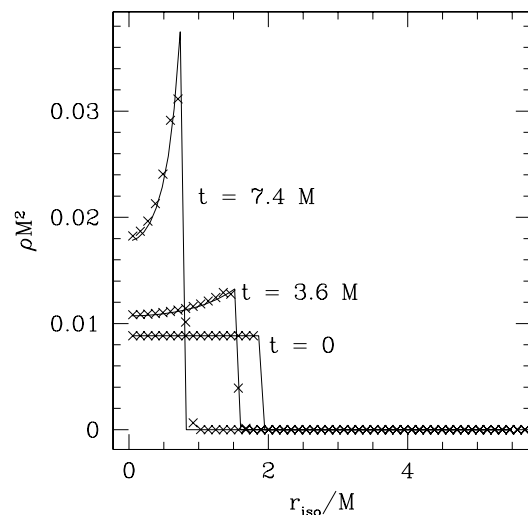


FIG. 5: The density, defined by Equation (7) as a function of isotropic radius during Oppenheimer-Snyder collapse. We compare our numerical results (crosses) with the analytic profiles.

VIII. SINGLE STARS

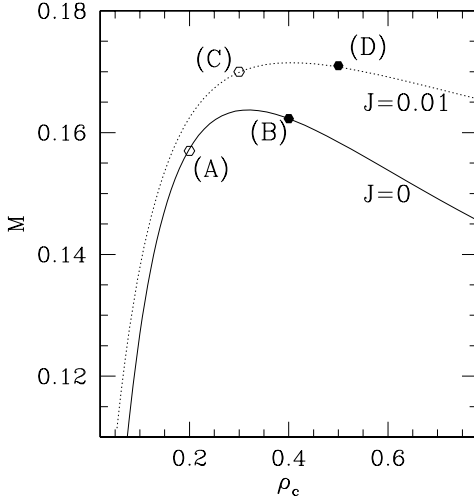
In this Section we study isolated stars, both non-rotating and rotating. The initial data, constructed from the OV solution for non-rotating equilibrium stars and with the code of [55] for equilibrium rotating stars, are summarized in Table I (see also Fig. 6.) We use the same coordinates as used in [55] (except transformed from spherical to Cartesian). For spherical, nonrotating systems (OV stars), these are the familiar isotropic coordinates. In these coordinates, the 3-metric for stationary spherical stars is conformally flat, and the event horizon of a Schwarzschild black hole is located at $r = 0.5M$. All stars are $n = 1$, $\Gamma = 2$ polytropes (see Eq. (23)), and are dynamically evolved with the gamma-law equation of state (22). The nondimensional units throughout are set by requiring $\kappa = G = c = 1$.

A. Static Stars

The stability properties of non-rotating $\Gamma = 2$ polytropes are known analytically and can be used as a test of our code. We use the OV [56] solution describing equilibrium polytropes in spherical symmetry as initial data, and evolve the matter and fields dynamically. An OV star is characterized by one parameter, which can be taken to be the central rest density ρ_c . (We will henceforth drop the subscript “0” on the rest density when referring to central rest density.) Along the sequence of increasing ρ_c , the mass M takes a maximum value M_{max} at a critical central density ρ_c^{crit} . (See Fig. 6.) Stars with $\rho_c < \rho_c^{\text{crit}}$ are dynamically stable, while stars with

TABLE I: Isolated Equilibrium Star Configurations ($\Gamma = 2$).

Star	M^a	M_0^b	R_{eq}^c	R_c^d	J/M^{2e}	$T/ W ^f$	Ω_c/Ω_{eq}^g	R_{pe}^h
A	0.157	0.171	0.700	0.866	0.00	0.000		1.00
B	0.162	0.178	0.540	0.714	0.00	0.000		1.00
C	0.170	0.186	0.697	0.881	0.35	0.032	1.00	0.88
D	0.171	0.187	0.596	0.780	0.34	0.031	1.00	0.87
E	0.279	0.304	1.251	1.613	1.02	0.230	2.44	0.30
F	0.049	0.050	1.240	1.290	0.72	0.053	5.88	0.75

^a ADM mass^b rest (baryonic) mass^c coordinate equatorial radius^d areal radius at the equator^e ratio of angular momentum to M^2 ^f ratio of kinetic to gravitational potential energy^g ratio of central to equatorial angular velocity^h ratio of polar to equatorial coordinate radiusFIG. 6: Stars A, B, C, and D and the constant- J sequences on which they lie. Open circles represent stable configurations, and closed circles denote unstable configurations.

$\rho_c > \rho_c^{\text{crit}}$ are unstable and collapse to black holes on a dynamical timescale. The dynamical timescale is given by the free-fall time $\rho_c^{-1/2}$. To verify that our code can distinguish stable and unstable configurations we evolve two very similar models on either side of the critical point at ρ_c^{crit} .

In our units, $\rho_c^{\text{crit}} = 0.32$ and $M_{\text{max}} = 0.164$. Star A has an initial central rest density $\rho_{ci} = 0.2$ and is therefore stable. We set our outer boundaries at $x, y, z = 2$ and evolve this star with three different resolutions 16^3 , 32^3 , and 64^3 , once again utilizing octant symmetry. In Fig. 7, we show the central density evolution for the three resolutions using harmonic slicing and the Gamma-driver shift. We see that our code does converge to the exact (stationary) solution. There are three sources of the deviations from exact second-order convergence (see also

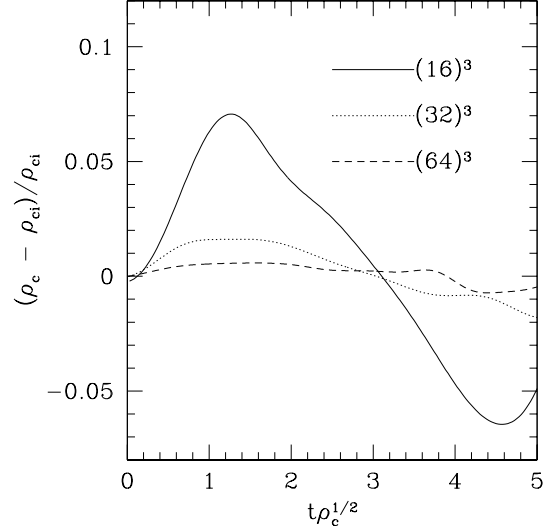
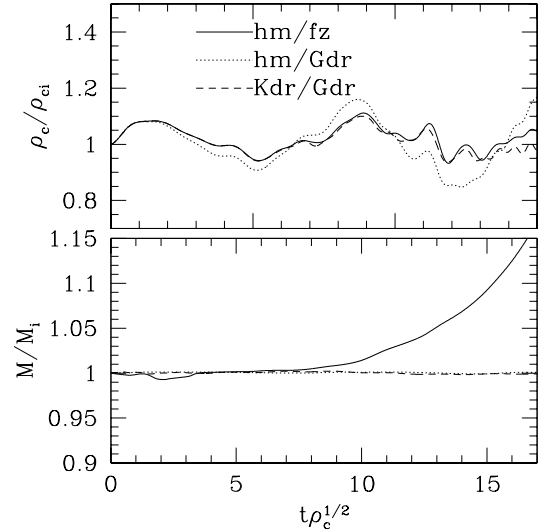


FIG. 7: Fractional change in the central rest density of star A when evolved on grids of three different resolutions.

FIG. 8: Star A evolved on a 32^3 grid using various gauge choices. Here “hm” refers to harmonic lapse, “Kdr” to K-driver lapse, “fz” to frozen shift, and “Gdr” to Gamma-driver shift.

[48]). First, there are components of the error which scale with a higher power of the grid width (e.g. Δx^3). Second, there is the noise caused by discontinuities at the surface of the star. Finally, errors are generated by imposing outer boundary conditions at finite distance.

In Fig. 8, we evolve on a 32^3 grid with several gauge choices. Already we see that the choice of gauge is important. Even in this static case, where the shift in the OV solution vanishes, it is necessary to use a dynamic shift for long term stability. With the Gamma-driver, we

evolve to $t\rho_c^{1/2} = 50$ ($t/M = 712$) and never encounter an instability.

We have stably evolved star A on the 32^3 grid for many fundamental radial oscillation periods, which have a period of about $\tau_r = 7\rho_c^{-1/2}$ [17]. However, we find that high-frequency, high-amplitude oscillations appear in ρ_c after a few periods and persist thereafter. The onset of these oscillations can be delayed and their amplitude diminished by increasing grid resolution. They can be removed altogether by making the hydrodynamic algorithm more implicit, i.e. by increasing the weight on the new timestep in the corrector step (See Section III B). This adversely affects our ability to handle shocks, though. The problem may also be resolved by the use of a more sophisticated hydrodynamics scheme (see, e.g., [26, 27]). For the less relativistic stellar model used by [26, 27] our code produces non-physical high-frequency oscillations after about 6 radial oscillation periods τ_r . For the applications discussed here, these late-time problems are not relevant.

Star B has an initial central density of $\rho_{ci} = 0.4$ and is dynamically unstable. We evolve this star with harmonic lapse and frozen shift, imposing outer boundaries at $1.2 M$, on two different grids (32^3 and 64^3). In Fig 9, we plot the central density and lapse as a function of time. The collapse is induced solely by the perturbations caused by putting the star on a discrete grid. Since these perturbations become smaller as grid resolution is increased, it is not surprising that the star on the lower-resolution grid collapses before the one on the higher-resolution grid. Since both collapse, it appears that 32^3 zones are sufficient to distinguish stable from unstable stars. Eventually, the star collapses to a point at which there are too few grid points across the star's diameter for the evolution to remain accurate. We terminate our evolutions when the error in the ADM mass exceeds 15% of the original mass. The 32^3 grid turns out to be too coarse for an apparent horizon to be located. We do locate an apparent horizon in the 64^3 run shortly before the simulation is terminated. At this point the central lapse has collapsed to $\alpha_c = 0.05$, and, as a measure of error, the ADM mass deviates by 10% from its initial value. The horizon mass agrees well with the ADM mass $M \approx M_{AH}$.

Also included in Fig. 9 is a 64^3 simulation using harmonic lapse and the Gamma-driver. Similar behavior is seen in these coordinates. We will investigate the performance of various gauge choices in more detail in the following Section.

B. Uniformly Rotating Stars

1. Inertial Frame

Simulations of systems with non-zero angular momentum are very sensitive to the choice of coordinates, which makes them very good test cases for comparing the nu-

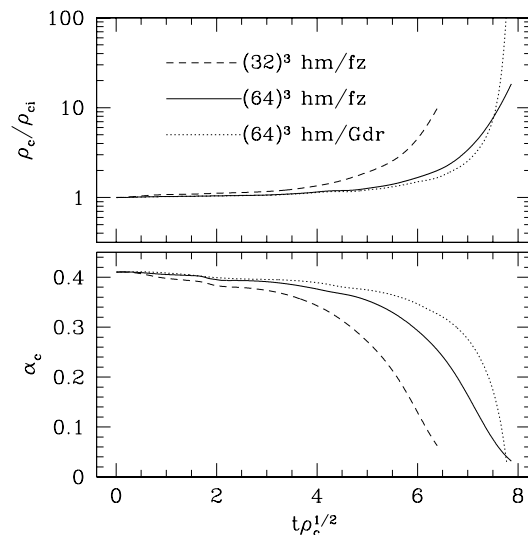


FIG. 9: The collapse of star B seen with various gauge choices. The abbreviations are the same as in Fig. 8.

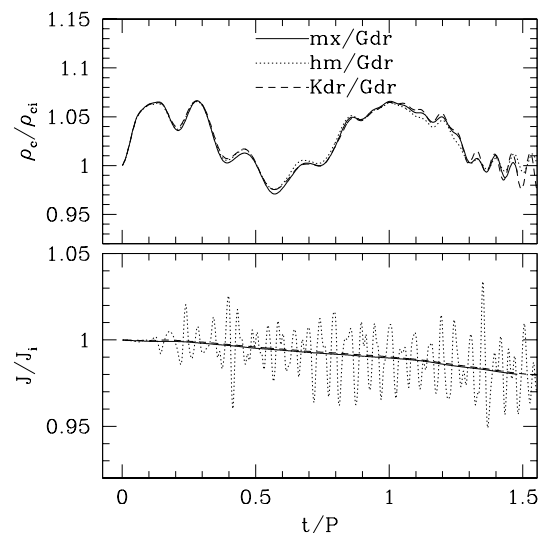


FIG. 10: Star C evolved on a 64×32^2 grid with Gamma-driver shift condition and various lapse choices.

merical behavior of different gauges and slicings. Most of these effects can be seen when we evolve uniformly rotating stars.

We consider two uniformly rotating stars, stars C and D, on one constant angular momentum sequence $J = 0.01$ (see Fig. 6). The $J = 0.01$ sequence has a turning-point at $\rho_c^{\text{crit}} = 0.4$, $M_{\text{max}} = 0.172$. For a sequence of uniformly rotating stars, this turning point marks the onset of secular, not dynamical, radial instability [57]. It is possible for a star on the secularly unstable branch to be stabilized temporarily if the star begins to rotate differentially, so that no instability will develop on

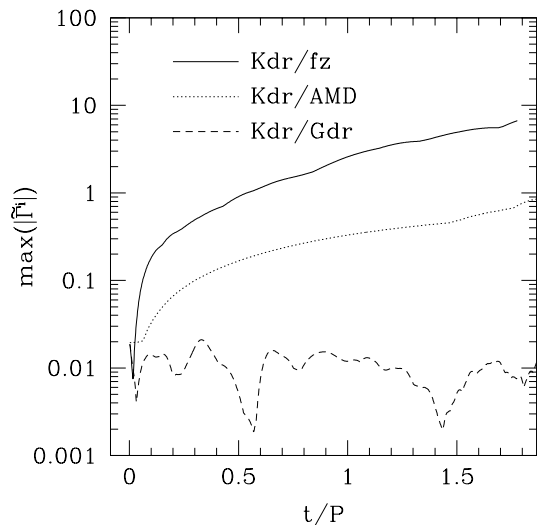


FIG. 11: Star C evolved on a 64×32^2 grid with K-driver lapse and various shift choices. We plot the maximum value of the absolute value of $\tilde{\Gamma}^x$ on the grid to show the dependence of $\tilde{\Gamma}^i$ on spatial gauge.

a dynamical timescale. However, prior numerical studies [24] have found the point of onset of dynamical instability to be very close to the point of onset of secular instability, which we confirm with our simulations here.

We again pick two similar stars on either side of the onset of secular instability: star C with $\rho_{ci} = 0.3$ on the stable branch and Star D with $\rho_{ci} = 0.5$ on the unstable branch. We dynamically evolve these two stars with different choices for the slicing and gauge. All simulations are performed on 64×32^2 grids, utilizing equatorial and π -symmetry to evolve only half of a hemisphere. The outer boundaries are placed at $[-1.5, 1.5] \times [0, 1.5]^2$. There are now two relevant timescales – the free-fall time $\tau_{ff} \sim \rho_c^{-1/2}$ and the orbital period P – and a reliable code must be able to stably evolve stable rotating stars for several of both timescales.

Results for star C with $\tau_{ff} = 1.83$ and $P = 26.38$ in our units, are plotted in Figures 10 and 11. In Fig. 10 we compare the evolution for maximal slicing (40), harmonic slicing (42) and the K-driver (41), all with the Gamma-driver shift condition (49). We find that there is little sensitivity to the lapse choice except for small oscillations in J which are only present with harmonic slicing.

In Fig. 11 where we compare the frozen shift condition (43), the AMD shift (44) and the Gamma driver (49), all evolved with the K-driver (41) for the lapse. This comparison demonstrates the great importance of choosing an appropriate shift condition for controlling $\tilde{\Gamma}^i$. AMD is dramatically better than frozen shift in this regard, and the Gamma driver is dramatically better than AMD. The behavior with AMD shift does not change significantly when the criteria for convergence of (44) is made stricter. Note that the modification (46) to AMD and

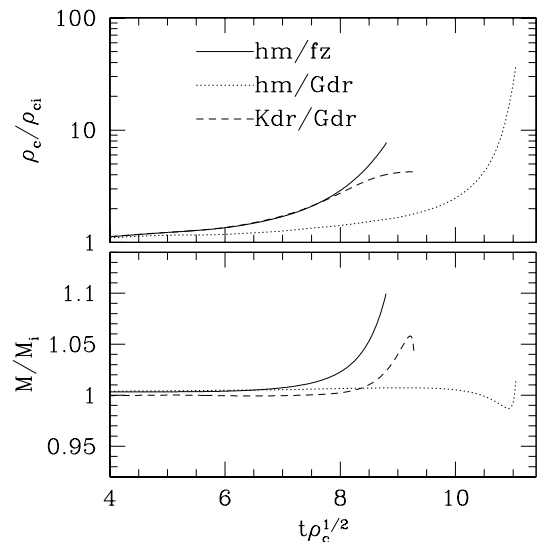


FIG. 12: The evolution of the central rest density and the ADM mass as star D collapses.

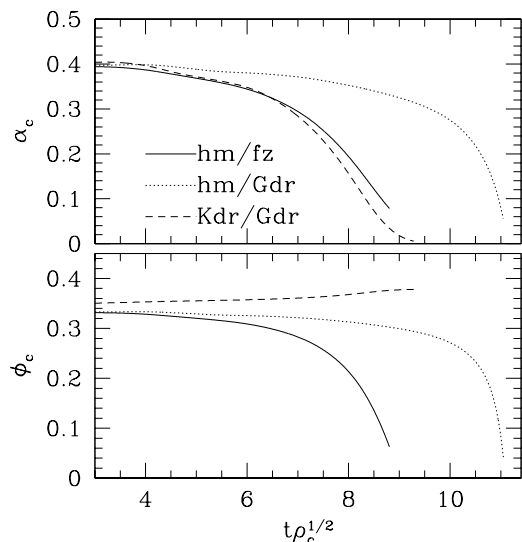


FIG. 13: The evolution of the lapse and conformal exponent at the origin as star D collapses.

the Gamma-driver is not activated for this application.

Figures 12 and 13 show the behavior of the radially unstable star D under different coordinate choices. Once again, perturbations are induced solely by the finite difference error of the grid. We terminate simulations when mass conservation is violated by 10% or the code crashes. The singularity avoidance property of the K-driver, which approximates maximal slicing, is manifest: with the lapse collapsing to very small values, the proper time between time slices at the star's center becomes very small, which effectively “freezes” all quantities there. With harmonic slicing, α decreases more slowly, and we are able to reach

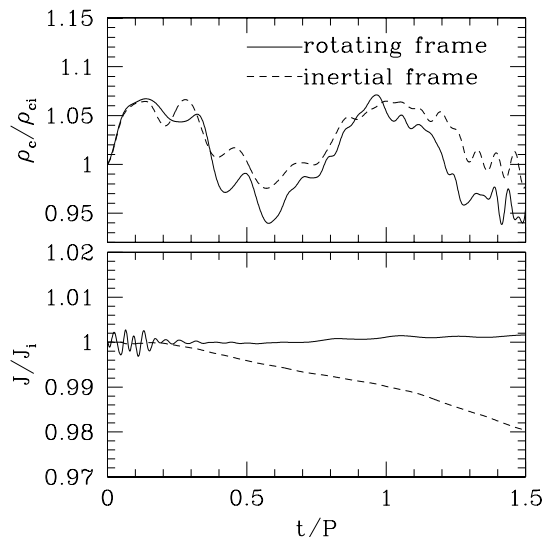


FIG. 14: The central density and angular momentum of star C evolved on a 64×32^2 grid with K-driver lapse and Gamma-driver shift in the inertial and in the rotating frame. We see that J is conserved much better in the rotating frame.

higher central densities, corresponding to later proper times, before the code crashes. Given their qualitatively different behavior, it is difficult to compare meaningfully the different lapse choices for this scenario. If one wants to see the central region reach the farthest stage of collapse before violation of mass conservation becomes unacceptable, harmonic lapse and Gamma-driver shift seem to be the optimal combination. One possible reason for this is the behavior of ϕ_c , the conformal exponent at the stellar center. For the gauge choices which are best suited to probing the central region, ϕ_c decreases significantly from its initial value. Inverting Shibata's reasoning for modifying the AMD gauge, we infer that this corresponds to choosing a gauge with infalling coordinates. This effectively increases the grid coverage of the collapsing star, resulting in a more accurate evolution.

We are only able to locate an apparent horizon in the harmonic lapse/Gamma driver simulation, and only in the last few timesteps, at which point $\alpha_c = 0.05$, $M_{AH}/M = 0.58$, and the error in ADM mass is about 2%. It seems that 64×32^2 zones are barely sufficient resolution for locating horizons for rotating stars reliably.

2. Rotating Frame

We compare results for uniformly rotating star C in the inertial frame to results in the corotating frame in Figure 14. In the corotating frame, $v^i = 0$ at $t = 0$. The light cylinder, where points of fixed coordinate label are moving on null paths, is at $r_{\text{cyl}} = 4.2$, well outside our outer boundaries at $x, y, z = 1.5$. All coordinate observers are therefore timelike everywhere on our grid. We see a dra-

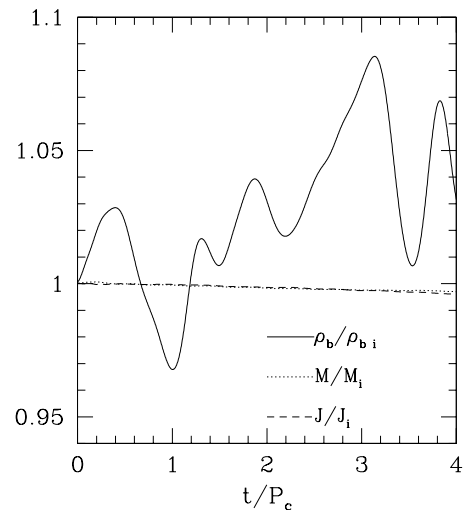


FIG. 15: Star E evolved for 4 central periods on a 64×32^2 grid. The M and J curves overlap.

matic improvement in angular momentum conservation in the corotating frame. This indicates that the loss of J in the inertial frame is caused by error in the advection of fluid quantities along v^i . Mass conservation is also better in the rotating frame, but not dramatically. Other quantities show qualitatively the same behavior in both frames. We have redone the evolution of collapsing star D with harmonic lapse and Gamma-driver shift in the corotating frame, and our results were almost identical to those in the inertial frame.

C. Differentially Rotating Stars

We now test the ability of our code to handle differential rotation. Differential rotation in neutron stars is relevant in several important astrophysical phenomena. Simulations in both Newtonian hydrodynamics [58] and full general relativity [21, 22] indicate that binary neutron star coalescence may well lead, at least temporarily, to a differentially rotating remnant, which can support significantly more rest mass than uniformly rotating stars [23]. Core collapse in a supernova may also result in a differentially rotating neutron star.

We construct axisymmetric equilibrium initial data, again following [55], with z chosen as the axis of symmetry. For the rotation profile, we choose

$$u^t u_\phi = R_{\text{eq}}^2 A^2 (\Omega_c - \Omega) \quad (67)$$

where Ω is the angular velocity of the fluid, Ω_c is the value of Ω on the rotation axis, R_{eq} is the equatorial coordinate radius, and A is a parameter that measures, in units of R_{eq} , the scale over which Ω changes. In the

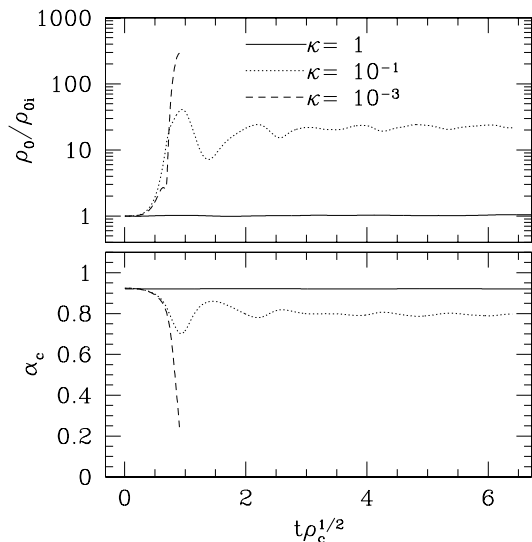


FIG. 16: The evolution of star F with 0%, 90%, and 99.9% of its pressure removed, respectively. When no pressure is removed, the star is stationary. When 90% is removed, we evolve until the new equilibrium is reached. When 99.9% is removed, the star collapses from an initial radius of $26M$ to a radius of $\sim 5M$, at which point the simulation becomes inaccurate, and we terminate it.

Newtonian limit this profile reduces to

$$\Omega = \frac{A^2 \Omega_c}{(x^2 + y^2)/R_{\text{eq}}^2 + A^2}. \quad (68)$$

For $A \rightarrow \infty$ one recovers uniform rotation.

In Fig. 15 we present results for star E with $\rho_{\text{max}} = 0.07$, $A^{-1} = 1$, $R_{\text{eq}}/M = 4.48$, $T/|W| = 0.23$, and $J/M^2 = 1.02$. This star's rest mass of $M_0 = 0.304$ exceeds the maximum allowed rest-mass of non-rotating $\Gamma = 2$ polytropes by 70 percent. We evolve this star on a 64×32^2 grid, using π -symmetry, with outer boundaries at $[-2, 2] \times [0, 2]^2$. The same star was evolved dynamically by [23], and we confirm their finding that the star is stable over several central rotation periods.

We found that simulations of differentially rotating stars are very sensitive tests of hydrodynamic advection schemes. In particular, when we used time-averaging instead of Crank-Nicholson to treat the advection terms (see Appendix A), we found that the angular momentum is conserved very poorly. The decrease in J also causes the central density to rise, and the numerical model to drift further and further from the true solution. This suggests that for differential rotation, the ability to successfully conserve angular momentum depends strongly on the finite difference algorithm used for the hydrodynamics.

In Fig. 16 we show results for star F, with $\rho_{\text{max}} = 0.0174$, $A^{-1} = 3$, $R_{\text{eq}}/M = 26.3$, $T/|W| = 0.0528$, and $J/M^2 = 0.715$. This model is identical to star I in Table II of [59], where this star was evolved in axisymmetry.

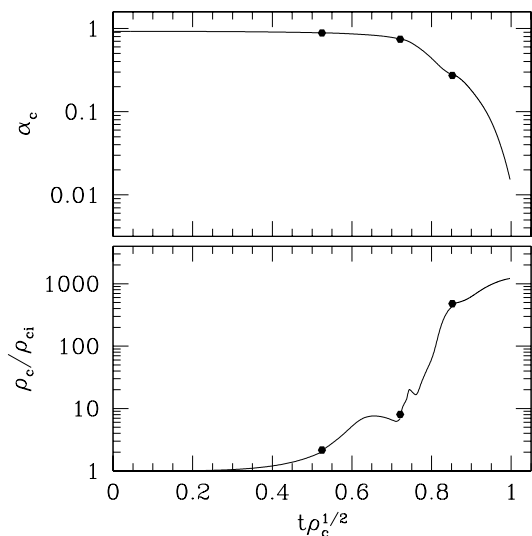


FIG. 17: The evolution of star F with 99.9% of its pressure removed. This time, we evolve on a $100^2 \times 50$ grid. The points mark times when the resolution was doubled.

Star F is radially stable, but, as in [59], we make the situation dynamic by depleting pressure from the star by artificially reducing the polytropic constant κ (which requires us to re-solve the Hamiltonian and momentum constraints). Removing pressure support causes the star to implode. For small depletion factors, this collapse is halted and the star bounces and finds a new, more compact, stable equilibrium configuration. When κ is reduced to a low enough value, the star will collapse to a black hole. Thus, there is a critical polytropic constant κ_{crit} separating these two outcomes. In [59], this critical value was found to be $\kappa_{\text{crit}} \approx 0.04$.

We evolve star F on a grid of $64^2 \times 32$ zones, with the outer boundaries located at 2, or equivalently $40.8 M$. In this simulation we use only equatorial symmetry so that non- π -symmetric perturbations can grow. We evolve three different cases; one without pressure depletion with $\kappa = 1$, a supercritical case with $\kappa = 0.1 > \kappa_{\text{crit}}$, and a subcritical case with $\kappa = 0.001 < \kappa_{\text{crit}}$. Both the second and third case present unique challenges. In the second case, the collapse is halted by a strong shock which must be handled accurately. In the third case, we must follow the collapse from a radius of $26.3M$ to a radius of $\approx M$. Our results are consistent with those of [59], even though our 3D simulations have a much poorer resolution than the axisymmetric simulations of [59]. In particular, our resolution is insufficient to follow the final stages of the $\kappa = 0.001$ collapse and prove that a black hole is formed.

In order to overcome this problem, we redo the $\kappa = 0.001$ collapse on a $100^2 \times 50$ grid. This grid is still too sparse to resolve a black hole with radius of approximately $1M$ if the outer boundaries are imposed at $40.8 M$. In order to resolve the black hole, we rezone our grid

several times during the implosion, halving the boundaries and halving the grid spacing, so that the total number of gridpoints remains constant (compare [60].) We present results for a simulation that was carried out on four different grids with outer boundaries at 2, 1.05 and 0.25. We use the K-driver and the Gamma-driver without the modification (46). With the modification, the functions $\tilde{\Gamma}^i$ grow very rapidly and cause the code to crash well before the radius reaches M . Turning off the modification means that ϕ_c will grow, and the coordinates will blow outward. We count on the grid rezoning to counter this effect. Also, we switch to frozen shift on the last and finest grid of the evolution. The Gamma-driver does not perform well on this segment, perhaps because the grid boundaries have been moved in to a point where the shift does not have its asymptotic form (58).

The results of this simulation are shown in Figs. 17 and 18. M and J remain within 10% of their initial values throughout (we terminate the calculation when this ceases to be true.) In our coordinates, the equatorial radius decreases from 1.24 ($25.3M$) to 0.04 ($0.8M$). Since ϕ_c is growing, part of this decrease in radius is a coordinate effect. The coordinate-independent circumferential radius at the equator (computed from $g_{\phi\phi}$) decreases from $27M$ to $1.7M$. At $t\rho_c^{1/2} = 0.98$, we locate an apparent horizon with surface area $\mathcal{A} = 0.0804$ corresponding to an “irreducible” mass of $(\mathcal{A}/16\pi^2)^{1/2} = 0.8M$. Is this a reasonable value? The existence of rotation and of mass outside of the black hole mean that we can no longer expect the irreducible mass of the hole to be equal to the ADM mass of the entire system. The area of the event horizon of a Kerr black hole with this system’s total M and J would be $\mathcal{A} = 0.109$. By breaking up the rest mass integral into pieces inside and outside the horizon, we find that 82% of the baryonic mass is inside the apparent horizon. If we assume that the values of M and J/M for the black hole are 82% of those of the total system, we arrive at the very crude estimate $\mathcal{A} = 0.0732$, which is within 10% of the value determined from the apparent horizon. We terminate our simulation $2.5M$ after the horizon is located, during which time its area does not change appreciably.

Our agreement with [59] indicates that nonaxisymmetric perturbations are not important in the collapse of this star. We confirm this in Fig. 18. As one can see, the density profiles remain axisymmetric throughout.

IX. BINARY SYSTEMS

Binary neutron stars are among the most promising sources of gravitational radiation for the new generation of gravitational wave interferometers. This makes the numerical simulation of such systems one of the most important goals of a fully relativistic hydrodynamics code and provides one of the most demanding tests for any such code. A binary system allows us to uncover potential problems that may not be evident in axisymmetric

scenarios. Previous simulations have focused on the coalescence and merger of binary neutron stars [21, 22]. In this Section we demonstrate that our code can stably evolve binaries in stable, quasi-circular orbits for over two periods (compare [17]).

As initial data for these simulations we adopt the data of [61, 62], describing two equal mass polytropes in co-rotating, quasi-circular orbit. These data have been constructed using the conformal “thin-sandwich” decomposition of the constraint equations [11, 12, 13, 63, 64] together with maximal slicing and spatial conformal flatness.

In this Section we focus on one particular case and postpone a more complete presentation for a forthcoming paper [30]. We model the neutron stars as $\Gamma = 2$ polytropes with an individual rest mass of $M_0^{\text{ind}} = 0.1$ in our nondimensional units (recall that the polytropic index κ is set to unity). At infinite separation, this corresponds to an individual gravitational mass of $M_\infty^{\text{ind}} = 0.096$. The compaction of $(M^{\text{ind}}/R)_\infty = 0.088$ implies that the gravitational fields are moderately relativistic (the maximum compaction for $\Gamma = 2$ polytropes is $(M^{\text{ind}}/R)_\infty = 0.21$). We adopt initial data for a binary separation of $z_a = 0.3$, where z_a is the ratio between the coordinate separation from the center of mass to the nearest point on the star’s surface to the farthest point (see [61, 62]), meaning that the separation between the stellar surfaces is about 86% of a stellar diameter. This separation is well outside the innermost stable circular orbit (ISCO) as determined by the analysis of initial data sets (see [65]). At this separation, the total binary ADM mass is $M = 0.19$ and the total angular momentum is $J/M^2 = 1.36$.

We evolve these initial data on three different grids. Two “small grid” simulations are evolved on 120×60^2 gridpoints, with a resolution of $\Delta x = \Delta y = \Delta z = 0.55M$ (the binary is symmetric across the equatorial plane, and, for equal mass stars, π -rotation symmetric around the center of mass). The individual stars are resolved by ≈ 16 gridpoints across the stellar diameter (compare [22] where much larger grids are used). One of these small grid evolutions is performed in the inertial frame, the other in a rotating frame. On these small (uniform) grids, the outer boundaries are imposed very close to the stars (at a separation of two stellar diameters), which we found to introduce numerical noise. We therefore repeated these simulations on a “large grid”, performed in a rotating frame, where we doubled the number of gridpoints and the separation to the outer boundary, while keeping the grid resolution constant. This corresponds to a numerical grid covering a cubic coordinate volume of side $[-66, 66]$ in the units of Fig. 19. The size of this grid is such that the corner points almost “touch” the surface of the light cylinder, the cylinder with coordinate radius $R_L = 1/\Omega$, where Ω is the rotating frame angular frequency. The possibility of evolution in rotating frames with grids that extend beyond the light cylinder will be studied in a future article.

We used Courant factors of 0.30 and 0.46 for the small

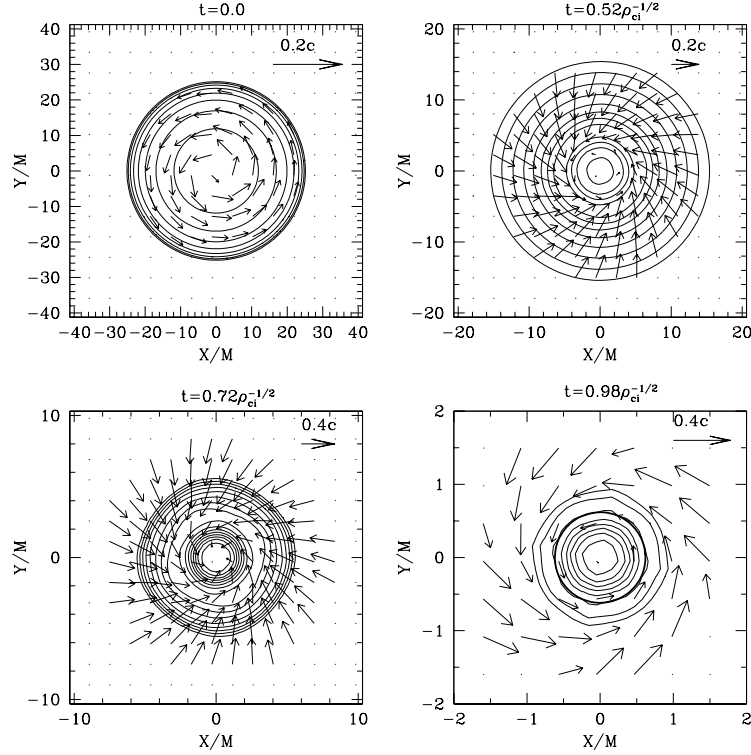


FIG. 18: Snapshots of the rest density contour lines for ρ_0 and the velocity field (v^x, v^y) in the equatorial plane for a simulation of the collapse of star F with $\kappa = 0.001$. The contour lines are drawn for $\rho_0 = 10^{-(0.2j+0.1)}\rho_0^{Max}$, where ρ_0^{Max} denotes the instantaneous maximum value of ρ_0 for $j = 0, 1, \dots, 7$. Vectors indicate the local velocity field, v^i . The thick circle on the last frame marks the apparent horizon.

and large grid runs respectively, resulting in about 3,000 timesteps per orbit for the latter case. We adopt the K-driver (41) for the lapse (using $\epsilon = 0.625$, $c = 0.1$, and 5 substeps per step) and the gamma-driver (49) for the shift (using $\eta = 0.2$, $k = 0.005$ and 10 substeps). We used Sommerfeld type boundary conditions for the gravitational fields (see Section II D) and *Copy* type for the hydrodynamical fields (see Section III E). For the artificial viscosity we used $C_{Qvis} = 0.1$ and $C_{Lvis} = 0$ (see Section III C). We also used $c_H = 0.050\Delta T$ in (19). For the small, rotating frame grid, this choice led to the code crashing after about a period and a half. However, we found that restarting the code just before that with $c_H = 0.024\Delta T$ allowed us to continue the evolution. For the large grid, no such adjustment was necessary. For both small grid simulations we found that the stars quickly drift apart. To compensate for this we reduced the orbital angular velocity Ω by 2% for these two cases. Again, no such adjustment was required for the large grid.

In Fig. 19 we show contour plots of the rest density ρ_0 at half period intervals for the large grid simulation in the rotating frame. The three-velocity of the fluid is represented by the arrows. The fact that the different panels look almost identical indicates how well the binary remains in its circular orbit.

Imposing the outer boundaries at smaller separations we were unable to keep the binary in circular orbit. In Fig. 20 we plot the coordinate separation d between the two points of maximum density for all three simulations as a function of time [67]. For the small grid in the inertial frame, the two stars start to drift apart after a very short time. When performing the same simulation in a rotating frame, the stars remained in binary orbit for about two periods, but ultimately merge. This merger is triggered by the development of orbital eccentricity. When we compare this small grid run with the large grid simulation, we see that the eccentricity is greatly reduced [68]. This result seems to validate the quasi-equilibrium approach to obtaining reasonable initial data for corotating neutron star binaries in circular equilibrium and underlies the importance of the boundary proximity effect in these simulations.

We show more diagnostics of these runs in Figs. 21 through 25. In Fig. 21 we show the rest mass M_0 , gravitational mass M , and angular momentum J for the three different simulations. The use of a rotating frame, which minimizes fluid advection through the numerical grid, leads to large improvements, especially in the conservation of angular momentum. Close outer boundaries lead to considerable numerical error. The system loses mass

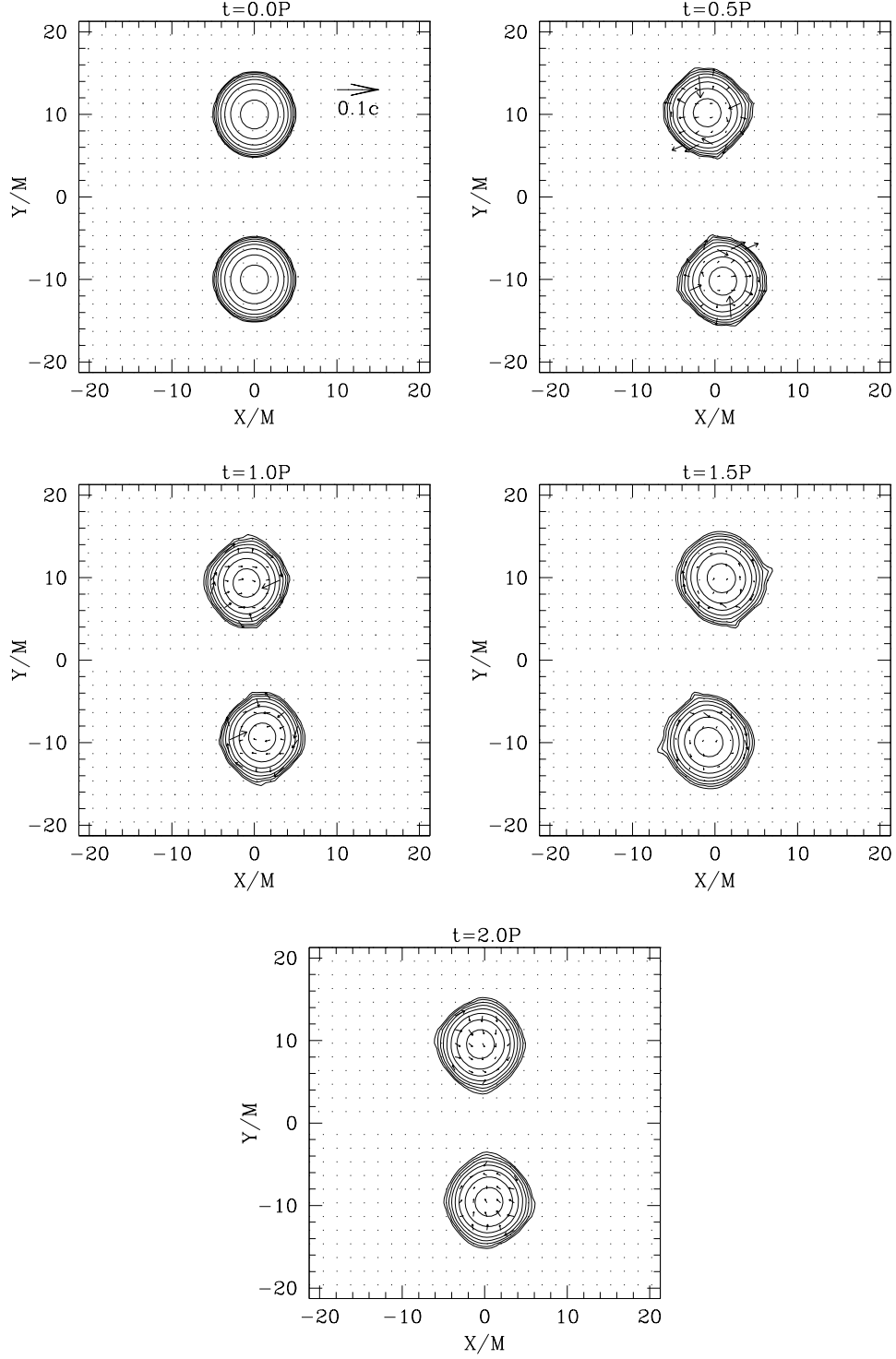


FIG. 19: Snapshots in a rotating coordinate frame of the rest density contour lines and the velocity field in the equatorial plane for a simulation of a corotating binary. The contour lines are drawn for $\rho_0 = 10^{-(0.2j+0.1)} \rho_{0i}^{Max}$, where ρ_{0i}^{Max} denotes the maximum value of the rest-density ρ_0 at $t = 0$ (here it is 0.0573), for $j = 0, 1, \dots, 7$. Vectors indicate the local velocity field and the scale is as shown in the top left-hand frame. The stars are orbiting clockwise in the *inertial* reference frame with an initial coordinate velocity of $0.102c$. P denotes the initial orbital period.

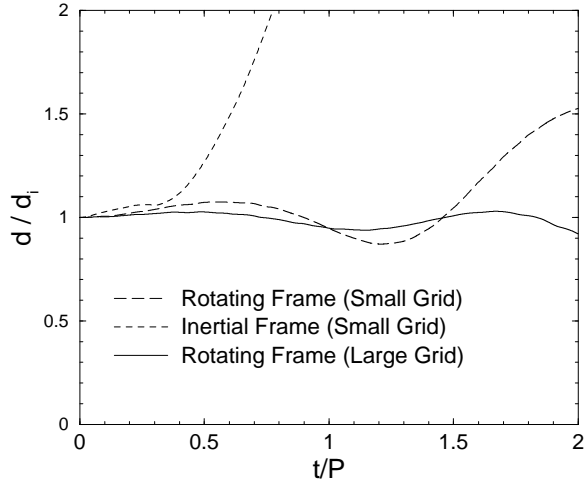


FIG. 20: Evolution of the coordinate separation between the maximum rest mass density points of the two stars in the binary system shown in Fig. 19. The time is given as a fraction of the initial orbital period and the separation d as a fraction of the initial value d_i .

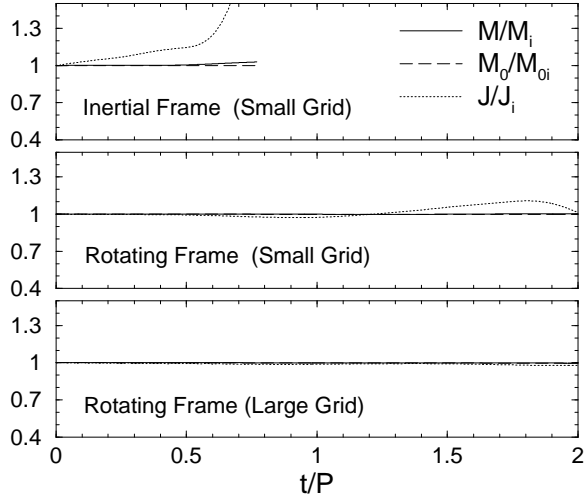


FIG. 21: Rest mass, gravitational mass, and angular momentum for the three simulations depicted in Fig. 20.

and angular momentum through the emission of gravitational radiation, but at a rate that should lead to smaller deviations than we find in our simulations. The maximum variation of the rest mass, gravitational mass, and angular momentum for the large grid run over the first two orbits was 0.3%, 0.3%, and 2.2% respectively.

In Fig. 22 we show the maximum rest density ρ_0 and the minimum value of the lapse α as functions of time. The small oscillations correspond to the fundamental mode of the individual neutron stars, which are induced by the truncation error of the finite grid resolution. The period of the oscillations $P \sim 16.6$ agrees well

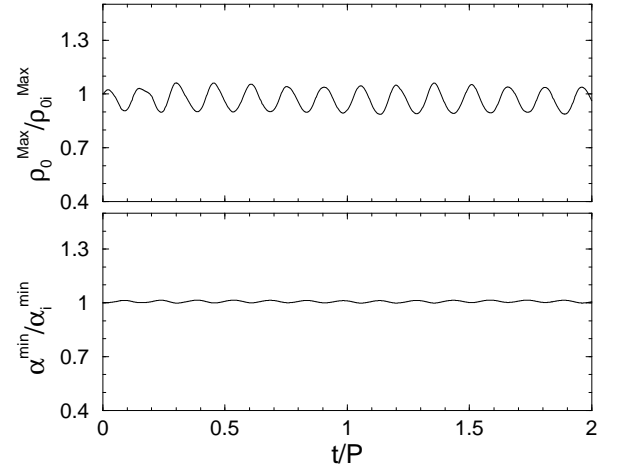


FIG. 22: Maximum rest mass density ρ_0 (top) and minimum lapse function α (bottom) for the large grid, rotating frame simulation.

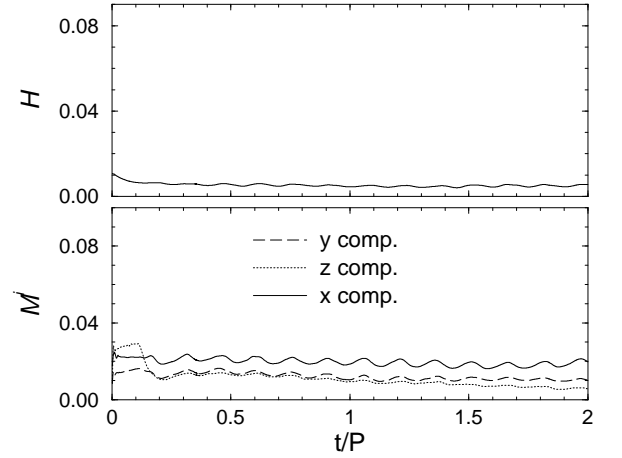


FIG. 23: L2 norms of the Hamiltonian constraint \mathcal{H} (top) and momentum constraints \mathcal{M}^i (bottom) for the large grid, rotating frame simulation. All the curves have been normalized as explained in Sec. IX.

with the theoretical value of $t = 16.0$ (see Fig. 32 in [17]). These oscillations are not damped, since for these runs we switched off the linear viscosity terms ($C_{\text{Lvis}} = 0$).

We monitor the Hamiltonian (16), momentum (17), and Gamma (18) constraints in Figs. 23-25. We show the L2 norms of the corresponding constraint violations. For the Hamiltonian and momentum constraint, these violations are normalized with respect to N_{HC} and N_{MC} evaluated at $t = 0$ (see Eqs. (59) and (60)). In Figure 24 we show the small-grid rotating-frame result for the Hamiltonian constraint violation (solid line), as well as the result from a similar small-grid evolution in which we set $c_H = 0$ (dashed line) [69]. The difference between these two lines illustrates the effect of the addition of this

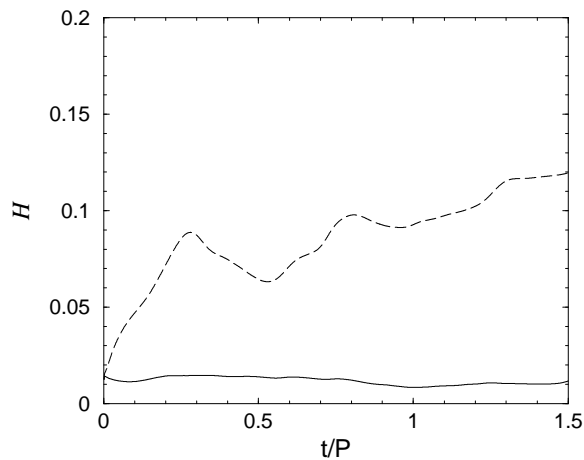


FIG. 24: L2 norms of the Hamiltonian constraint \mathcal{H} for small grid, rotating frame simulations, using $c_H = 0.05\Delta T$ (solid curve) and $c_H = 0.00$ (dashed curve). This plot shows the effect of the c_H term in Eq. (19) on the conservation of the Hamiltonian constraint.

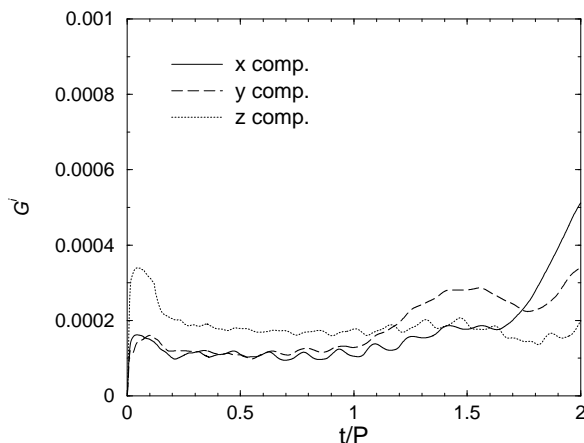


FIG. 25: L2 norm of the gamma constraint \mathcal{G}^i for the large grid, rotating frame simulation.

particular term in Eq. (19) on the conservation of the Hamiltonian constraint. The constraint deviations \mathcal{G}^i , which are particularly sensitive to the choice of spatial gauge, remain well behaved during the first two orbits in these evolutions with the Gamma-driver.

As this example demonstrates, our code is able to stably evolve binaries in stable, quasi-circular orbits for over two orbital periods. In a forthcoming paper [30] we will use this code to systematically study binary sequences, both to dynamically locate the ISCO and to test how accurately currently available quasi-equilibrium initial data represent binaries in quasi-circular orbit.

X. SUMMARY

We have tested our 3+1 relativistic hydrodynamics code on a variety of problems. We find that our current algorithm, supplemented by driver gauge conditions, is rather robust. The grid resources required for stable evolution and reasonable accuracy are modest. We accurately evolve shock tubes, spherical dust collapse, and relativistic spherical polytropes. We also evolved uniformly and differentially rotating equilibrium polytropes, and maintained stable configurations stationary for several rotational periods. Two applications carried out with our code are particularly significant. First, we examined the collapse from large radius of a star with significant spin to a Kerr black hole. Second, we evolved stable binary neutron stars for several orbits, maintaining quasi-circular equilibrium. The first application indicates that we can study the effects of angular momentum on gravitational collapse and on the resulting waveform, an effort already initiated in [59, 66]. The second application indicates that we can identify and evolve dynamically stable quasi-circular neutron star binaries. This ability can be used to locate the ISCO dynamically and to follow the transition from an inspiral to a plunge trajectory. In addition, dynamic simulation allows us to improve binary initial data, for example by allowing initial “junk” gravitational radiation to propagate away. We also hope to compute detailed gravitational waveforms from these binaries, refining the wavetrains reported in [49, 50].

We note that several challenges remain to be addressed before there exists a code capable of modeling all the gravitational wave sources of current astrophysical interest. One problem is the need to maintain adequate grid coverage of the collapsing star or inspiralling binary while still keeping the outer boundaries sufficiently distant, i.e. the problem of dynamic range. Adaptive mesh techniques far more sophisticated than the crude rezoning used here may be necessary. A related problem concerns gravitational wave extraction, as it currently is not possible to place outer boundaries in the wave zone. Finally, the formation of black hole singularities in hydrodynamic collapse scenarios remains an additional challenge to determining the late-term behavior of such systems. Special singularity-handling techniques, such as excision, need to be developed further.

Acknowledgments

It is a pleasure to thank Masaru Shibata and Hwei-Jang Yo for useful discussions. Most of the calculations were performed at the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign (UIUC). This paper was supported in part by NSF Grants PHY-0090310 and PHY-0205155 and NASA Grant NAG 5-10781 at UIUC and NSF Grant PHY 01-39907 at Bowdoin College.

APPENDIX A: TREATMENT OF ADVECTIVE TERMS IN THE HYDRODYNAMIC EQUATIONS

Solving Eqs. (26)-(28) requires solving equations of the form

$$\frac{\partial q}{\partial t} + \frac{\partial(qv)}{\partial x} = S \quad (\text{A1})$$

Let us finite difference this equation. Let q_i^n be the value of q at gridpoint i on time level n . Let Δx be the coordinate distance between neighboring gridpoints, and ΔT be the timestep. The Courant factor is $C = \Delta T/\Delta x$. Then we difference (A1), say for the predictor step, as

$$q_i^{n+1} = q_i^n + \Delta T[(v_{i-1/2}^n q_{i-1/2}^n - v_{i+1/2}^n q_{i+1/2}^n) + S_i^n] \quad (\text{A2})$$

where

$$v_{i+1/2}^n = \frac{\rho_{\star i}^n v_i^n + \rho_{\star i+1}^n v_{i+1}^n}{\rho_{\star i}^n + \rho_{\star i+1}^n} \quad (\text{A3})$$

$$q_{i+1/2}^n = \begin{cases} q_i^n + \Delta x \nabla_i^n q/2 & \text{if } v_i^n > 0 \\ q_{i+1}^n - \Delta x \nabla_{i+1}^n q/2 & \text{if } v_i^n < 0 \end{cases} \quad (\text{A4})$$

In (A4),

$$\nabla_i^n q = \begin{cases} \frac{2\Delta_{i-1/2}^n q \Delta_{i+1/2}^n q}{\Delta_{i-1/2}^n q + \Delta_{i+1/2}^n q} & \text{if } \Delta_{i-1/2}^n q \Delta_{i+1/2}^n q > 0 \\ 0 & \text{otherwise} \end{cases} \quad (\text{A5})$$

where $\Delta_{i+1/2}^n q = (q_{i+1}^n - q_i^n)/\Delta x$. In many van Leer schemes, the term $q + \Delta x \nabla_i^n q/2$ in Eq. (A4) is replaced by the time-averaging expression $q + (\Delta x - v\Delta T)\nabla_i^n q/2$. Since we use a predictor-corrector method, we don't need to time average.

APPENDIX B: NEWTONIAN LIMIT OF THE EULER EQUATION IN A ROTATING FRAME

Here we recover the Newtonian Euler equation in a rotating frame, as given for instance in [70], taking the weak-field limit of the general relativistic Euler equation. All variables and differential operators are given in the rotating frame, with the exception of "barred" quantities that reside in the inertial frame. For simplicity, we will work with the general relativistic Euler equation written as a function of the variable \hat{u}_k , defined as

$$\hat{u}_k \equiv hu_k = \hat{\gamma}_{ki} u^0 h e^{4\phi} (v^i + \beta^i). \quad (\text{B1})$$

Accordingly, we have

$$\begin{aligned} \partial_t(\rho_\star \hat{u}_k) + \partial_i(\rho_\star \hat{u}_k v^i) = & \\ -\alpha e^{6\phi} \partial_k P - \rho_\star u^0 \alpha h \partial_k \alpha - \rho_\star \hat{u}_j \partial_k \beta^j & \\ -2 \frac{\rho_\star h ((u^0 \alpha)^2 - 1)}{u^0} \partial_k \phi + \frac{\rho_\star e^{-4\phi} \hat{u}_i \hat{u}_j}{2u^0 h} \partial_k \hat{\gamma}^{ij}, & \end{aligned} \quad (\text{B2})$$

where ρ_\star and h are defined in Sec. (III A). Using the continuity equation (24) we can re-write the lhs of equation (B2) as

$$\partial_t(\rho_\star \hat{u}_k) + \partial_i(\rho_\star \hat{u}_k v^i) = \rho_\star (\partial_t \hat{u}_k + v^i \partial_i \hat{u}_k). \quad (\text{B3})$$

To obtain the Newtonian limit, we expand the different terms in equation (B2) to first order in the Newtonian potential ϕ_N and the square of the fluid velocity v . For instance:

$$\begin{aligned} \hat{\gamma}_{ki} &\rightarrow \delta_{ki} \\ u^0 &\rightarrow 1 + \frac{v^2}{2} \\ h &\rightarrow 1 \\ e^\phi &\rightarrow 1 - 2\phi_N \\ \alpha &\rightarrow 1 + \phi_N \\ \beta^i = \bar{\beta}^i + (\vec{\Omega} \times \vec{r})^i &\rightarrow (\vec{\Omega} \times \vec{r})^i, \end{aligned} \quad (\text{B4})$$

where $\bar{\beta}^i$ is the shift vector in a inertial frame, $\vec{\Omega}$ is the angular velocity of the rotating frame with respect to the inertial frame, and $\vec{r} \equiv (x, y, z)$ is the position vector. The limits (B4) combined with equation (B1) give

$$\hat{u}_k \rightarrow v^k + (\vec{\Omega} \times \vec{r})^k \quad (\text{B5})$$

We proceed now to take the Newtonian limit of the rhs of Eq. (B3). To do so, we note that

$$\partial_t \hat{u}_k \rightarrow \partial_t v^k + \partial_t (\vec{\Omega} \times \vec{r})^k = \partial_t v^k \quad (\text{B6})$$

since $\partial_t \vec{\Omega} = \partial_t \vec{r} = 0$, and

$$\begin{aligned} v^i \partial_i \hat{u}_k &\rightarrow v^i \partial_i (v^k + (\vec{\Omega} \times \vec{r})^k) \\ &= v^i \partial_i v^k + (\vec{\Omega} \times \vec{v})^k. \end{aligned} \quad (\text{B7})$$

These conditions together with equation give (B3) give the Newtonian limit of the lhs of equation (B2)

$$\begin{aligned} \partial_t(\rho_\star \hat{u}_k) + \partial_i(\rho_\star \hat{u}_k v^i) &\rightarrow \\ \rho_\star (\partial_t v^k + v^i \partial_i v^k + (\vec{\Omega} \times \vec{v})^k). \end{aligned} \quad (\text{B8})$$

To work on the rhs of equation (B2) to Newtonian order, we derive the following limits using Eqs. (B4), again keeping only the first order terms in ϕ_N and v^2 :

$$\begin{aligned} \alpha e^{6\phi} \partial_k P &\rightarrow \partial_k P \\ \rho_\star u^0 \alpha h \partial_k \alpha &\rightarrow \rho_\star \partial_k \phi_N \\ 2\rho_\star h \frac{((u^0 \alpha)^2 - 1)}{u^0} \partial_k \phi &\rightarrow 0 \\ \frac{\rho_\star e^{-4\phi}}{2u^0 h} \hat{u}_i \hat{u}_j \partial_k \hat{\gamma}^{ij} &\rightarrow 0. \end{aligned} \quad (\text{B9})$$

The final term is composed using (B5):

$$\rho_\star \hat{u}_j \partial_k \beta^j \rightarrow \rho_\star (v^j + (\vec{\Omega} \times \vec{r})^j) \partial_k (\vec{\Omega} \times \vec{r})^j. \quad (\text{B10})$$

We rewrite the first term above as

$$\begin{aligned}\rho_\star v^j \partial_k (\vec{\Omega} \times \vec{r})^j &= \rho_\star v^j (\vec{\Omega} \times \partial_k \vec{r})^j \\ &= \rho_\star \epsilon_{k j n} v^j \Omega_n \\ &= -\rho_\star (\vec{\Omega} \times \vec{v})^k, \end{aligned} \quad (\text{B11})$$

and the second term as

$$\rho_\star (\vec{\Omega} \times \vec{r})^j \partial_k (\vec{\Omega} \times \vec{r})^j = -\rho_\star \left(\vec{\Omega} \times (\vec{\Omega} \times \vec{r}) \right)^k. \quad (\text{B12})$$

Combining (B8), (B9), (B11), and (B12), yields

$$\begin{aligned}\rho_\star \left(\partial_t v^k + v^i \partial_i v^k + (\vec{\Omega} \times \vec{v})^k \right) &= \\ -\partial_k P - \rho_\star \partial_k \phi_N - \rho_\star (\vec{\Omega} \times \vec{v})^k & \quad (\text{B13}) \\ -\rho_\star \left(\vec{\Omega} \times (\vec{\Omega} \times \vec{r}) \right)^k. & \end{aligned}$$

Rearranging terms and replacing ρ_\star by its limit the mass density ρ yields the Newtonian limit of the general relativistic Euler equation (B2) :

$$\begin{aligned}\partial_t v^k + v^i \partial_i v^k &= \\ -\frac{1}{\rho} \partial_k P - \partial_k \phi_N - 2(\vec{\Omega} \times \vec{v})^k - \left(\vec{\Omega} \times (\vec{\Omega} \times \vec{r}) \right)^k, & \quad (\text{B14})\end{aligned}$$

where the last two terms of the rhs correspond to the familiar Coriolis and centrifugal force terms.

APPENDIX C: ADM MASS AND ANGULAR MOMENTUM IN ROTATING FRAMES

In this Appendix, the “barred” fields represent variables in the inertial frame, while the non-barred ones are quantities in rotating frames. In Sec. V, we defined

the total mass and angular momentum of an asymptotically flat spacetime by two surface integrals (Eqs. (61) and (62) respectively) which characterize the asymptotic behavior of the metric on a time slice. These surface integrals were transformed into the volume integrals (63) and (64) according to the calculation described in [33]. These volume integrals are numerically evaluated in our code on the computational grid. When working in rotating frames, one might worry that these integrals do not apply, since the 4-metric is not asymptotically flat due to the $\vec{\Omega} \times \vec{r}$ term in the shift. It turns out that this is not a problem, since the surface integral formulae for M and J can be obtained assuming only that the 3-metric and extrinsic curvature are asymptotically flat [71]. Therefore, we can evaluate the volume integrals (63) and (64) in the rotating frame and be sure that the M and J that we find at a given time will be the same as what we would have found by transforming into an inertial frame and then computing the integrals. We can see this explicitly by transforming the integrands from an inertial to a rotating frame. For example, the mass (63) written in terms of the “barred” inertial frame quantities is

$$\begin{aligned}M &= \int_V \left(e^{5\bar{\phi}} (\bar{\rho} + \frac{1}{16\pi} \bar{A}_{ij} \bar{A}^{ij} - \frac{1}{24\pi} \bar{K}^2) \right. \\ &\quad \left. - \frac{1}{16\pi} \bar{\Gamma}^{ijk} \bar{\Gamma}_{jik} + \frac{1 - e^{\bar{\phi}}}{16\pi} \bar{R} \right) d^3 \bar{x}. \end{aligned} \quad (\text{C1})$$

For simplicity, we take the inertial coordinate system to be the one which is instantaneously aligned with our rotating frame at the time that we are computing M and J . Then the transformation is given by Eqs. (51), (52), and (53). (From Eq. (7), we see that ρ is an invariant.) Applying these rules, we see that every term in the integrand is identical in the inertial and rotating frames. The same is true of the integrands for J and M_0 .

-
- [1] C. Cutler *et. al.*, Phys. Rev. Lett. **70**, 2984 (1993).
 - [2] E. E. Flanagan and S. A. Hughes, Phys. Rev. D **57**, 4566 (1998).
 - [3] A. Buonanno and T. Damour, Phys. Rev. D **62**, 064015 (2000).
 - [4] R. Narayan, B. Paczynski, and T. Piran, Astrophys. J. **395** L83, (1992).
 - [5] T. Piran, in *Proceedings of GR16*, (World Scientific, Singapore, in press) (also gr-qc/0205045).
 - [6] Ya. B. Zel’dovich and M. A. Podurets, Sov. Astron. **18**, 17 (1965); S. L. Shapiro and S. A. Teukolsky, Astrophys. J. Lett. **292**, L41 (1985); G. D. Quinlan and S. L. Shapiro, Astrophys. J. **343**, 725 (1989).
 - [7] S. Balberg and S. L. Shapiro, Phys. Rev. Lett. **88**, 101301 (2002).
 - [8] M. J. Rees, in *Black Holes in Binaries and Galactic Nuclei*, eds. L. Kaper, E. P. J. van den Heuvel, and P. A. Woudt (New York: Springer-Verlag, 2001); M. Saijo, T. W. Baumgarte, S. L. Shapiro, and M. Shibata, Astrophys. J. **569**, 349 (2002); M. Shibata and S. L. Shapiro, Astrophys. J. Lett. **572**, L39 (2002).
 - [9] M. M. May and R. H. White, Phys. Rev. D **141**, 1232 (1966).
 - [10] J. A. Font, Living Rev. Rel. **3** (2000) 2.
 - [11] J. R. Wilson and G. J. Mathews, Phys. Rev. Lett. **75**, 4161 (1995).
 - [12] J. R. Wilson, G. J. Mathews, and P. Marronetti, Phys. Rev. D **54**, 1317 (1996).
 - [13] J. R. Wilson and G. J. Mathews, G. J., in *Frontiers in Numerical Relativity* ed. C. R. Evans, L. S. Finn & D. W. Hobill, (Cambridge University Press, Cambridge, 1989).
 - [14] E. Flanagan, Phys. Rev. Lett., **82**, 1354 (1999).
 - [15] G. J. Mathews and J. R. Wilson, Phys. Rev. D, **61**, 127304 (2000).
 - [16] J. R. Wilson, Astrophys. J. **173**, 431 (1972).
 - [17] M. Shibata, Phys. Rev. D **60**, 104052 (1999).
 - [18] M. Shibata and T. Nakamura, Phys. Rev. D **52**, 5428 (1995).
 - [19] B. van Leer, J. Comput. Phys. **23**, 276 (1977).
 - [20] K. Oohara and T. Nakamura, Prog. Theor. Phys. **82**, 535

- (1989).
- [21] M. Shibata and K. Uryū, Phys. Rev. D **61**, 064001 (2000).
 - [22] M. Shibata and K. Uryū, Prog. Theor. Phys. **107**, 265 (2002).
 - [23] T. W. Baumgarte, S. L. Shapiro, and M. Shibata, Astrophys. J. **528**, L29 (2000).
 - [24] M. Shibata, T. W. Baumgarte, and S. L. Shapiro, Phys. Rev. D **61**, 044012 (2000).
 - [25] M. Shibata, T. W. Baumgarte, and S. L. Shapiro, Astrophys. J. **542**, 453 (2000).
 - [26] J. A. Font, M. Miller, W. Suen, and M. Tobias Phys. Rev. D **61**, 044011 (2000).
 - [27] J. A. Font *et. al.* Phys. Rev. D **65**, 084024 (2002).
 - [28] T. W. Baumgarte and S. L. Shapiro, Phys. Rev. D, **59**, 024007 (1999).
 - [29] F. D. Swesty, E. Y. Wang and A. C. Calder, Astrophys. J. **541**, 937 (2000).
 - [30] P. Marronetti, T. W. Baumgarte, M. D. Duez, and S. L. Shapiro, In preparation.
 - [31] R. Arnowitt, S. Deser and C. W. Misner, in *Gravitation: An Introduction to Current Research*, edited by L. Witten (Wiley, New York, 1962).
 - [32] S. A. Teukolsky, Phys. Rev. D **61** 087501 (2000).
 - [33] H.-J. Yo, T. W. Baumgarte and S. L. Shapiro, Phys. Rev. D, in press (2002).
 - [34] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes* (Cambridge University Press, Cambridge, England, 1992), p. 847.
 - [35] G. Yoneda and H. a. Shinkai, arXiv:gr-qc/0204002.
 - [36] J. M. Stone and M. L. Norman, Astrophys. J. Suppl. **80**, 753 (1992).
 - [37] W. C. Moss and J. W. White, J. Comput. Phys. **106**, 397 (1993).
 - [38] We actually use $\tilde{S}_k \equiv hS_k$ as our variable, where h is the enthalpy. In this section, we will ignore this point for the sake of simplicity.
 - [39] J. Balakrishna *et. al.*, Class. Quantum Grav. **13**, L135 (1996).
 - [40] M. Shibata, Prog. Theor. Phys. **101**, 1199 (1999).
 - [41] L. L. Smarr and J. W. York, Jr., Phys. Rev. D **17**, 2529 (1978).
 - [42] M. Alcubierre and B. Brügmann, Phys. Rev. D **63**, 104006 (2001).
 - [43] B. Carter, in *Active Galactic Nuclei*, eds. C. Hazard and S. Mitton (Cambridge Univ. Press, Cambridge, 1979), p.273.
 - [44] M. Saijo, M. Shibata, T. W. Baumgarte, and S. L. Shapiro, Astrophys. J. **548**, 919 (2001).
 - [45] T. W. Baumgarte, G. B. Cook, M. A. Scheel, S. L. Shapiro, and S. A. Teukolsky, Phys. Rev. D **54**, 4849 (1996).
 - [46] M. Parashar and J. C. Brown, in *Proceedings of the International Conference for High Performance Computing*, eds. S. Sahni, V. K. Prasanna and V. P. Bhatkar (Tata McGraw-Hill, New York, 1995), also www.caip.rutgers.edu/~parashar/DAGH/.
 - [47] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, "Efficient Management of Parallelism in Object Oriented Numerical Software Libraries", in *Modern Software Tools in Scientific Computing*, edited by E. Arge, A. M. Bruaset and H. P. Langtangen, p. 163, Birkhauser Press, 1997; see also the PETSc home page <http://www.mcs.anl.gov/petsc>.
 - [48] T. W. Baumgarte, S. A. Hughes, and S. L. Shapiro, Phys. Rev. D **60**, 087501 (1999).
 - [49] M. D. Duez, T. W. Baumgarte, and S. L. Shapiro, Phys. Rev. D, **63**, 084030 (2001).
 - [50] M. D. Duez, T. W. Baumgarte, S. L. Shapiro, M. Shibata, and K. Uryū, Phys.Rev. D **65** 024016 (2002).
 - [51] K. W. Thompson, J. Fluid Mech. **171**, 365 (1986).
 - [52] M. L. Norman and K.-H. Winkler, in *Radiation Hydrodynamics*, ed. M. L. Norman and K.-H. Winkler (Dordrecht, Reidel, 1983).
 - [53] J. R. Oppenheimer and H. Snyder, Phys. Rev. **56**, 455 (1939).
 - [54] L. I. Petrich, S. L. Shapiro, and S. A. Teukolsky, Phys. Rev. D **31** 2459 (1985).
 - [55] G. B. Cook, S. L. Shapiro, and S. A. Teukolsky, Astrophys. J. **398**, 203 (1992).
 - [56] J. R. Oppenheimer and G. Volkoff, Phys. Rev. **55**, 374 (1939).
 - [57] J. L. Friedman, J. R. Ipser, and R. D. Sorkin, Astrophys. J. **325**, 722 (1988).
 - [58] F. A. Rasio and S. L. Shapiro, Astrophys. J. **401**, 226 (1992); **432**, 242 (1994); **438**, 887 (1995); Class. Quant. Grav. **16** R1 (1999).
 - [59] M. Shibata, Prog. Theor. Phys. **104** 325 (2000)
 - [60] M. Shibata and S. L. Shapiro, Astrophys. J. **572**, L39 (2002).
 - [61] T. W. Baumgarte, G. B. Cook, M. A. Scheel, S. L. Shapiro, and S. A. Teukolsky, Phys. Rev. Lett. **79**, 1182 (1997).
 - [62] T. W. Baumgarte, G. B. Cook, M. A. Scheel, S. L. Shapiro, and S. A. Teukolsky, Phys. Rev. D **57**, 7299 (1998).
 - [63] J. W. York, Jr., Phys. Rev. Lett. **82**, 1350 (1999).
 - [64] G. B. Cook, Living Rev. Rel. **5**, 1 (2000).
 - [65] T. W. Baumgarte, G. B. Cook, M. A. Scheel, S. L. Shapiro, and S. A. Teukolsky, Phys. Rev. D **57**, 6181 (1998).
 - [66] T. Nakamura, Prog. Theor. Phys. **65**, 1876 (1981).
 - [67] We define the coordinate separation by $d = (\int_V d^3x f(\rho_\star r)) / (\int_V d^3x f(\rho_\star))$, where $f(x) = x$ if ρ_0 is greater than 0.01 times the maximum value of ρ_0 on the grid, and $f(x) = 0$ otherwise.
 - [68] T. Mora and C. M. Will (gr-qc/0208089) show that a small eccentricity may be present in some binary black hole initial data sets. Our results seem to indicate that for our binary neutron star simulations, the dominant source of spurious eccentricity is an insufficiently distant outer boundary.
 - [69] This comparison run was performed in a small grid due to the huge computational resources consumed by large grid runs. However, the difference between the two curves is similar to the one obtained when comparing two small grid runs.
 - [70] J. Tassoul, *Theory of Rotating Stars* (Princeton University Press, Princeton, 1978), p. 54.
 - [71] N. O. Murchadha and J. W. York, Jr, Phys. Rev. D **10**, 2345 (1974); J. M. Bowen and J. W. York, Jr, Phys. Rev. D **21**, 2047 (1980).